robotron

RVS-Rechnersystem K 1840

Diagnosehandbuch

Peripherie

VEB Robotron-Vertrieb Berlin

401

A003917

RVS-Rechnersystem K 1840

Diagnosehandbuch

Peripherie

VEB Robotron-Vertrieb Berlin

Diese Unterlage ist Bestandteil der Gesamtdokumentation "RVS-Rechnersystem K 1840".

Weitere Unterlagen siehe Nutzerhandbuch Peripherie und Betriebsdokumentation der peripheren Geräte.

Die vorliegende Dokumentation entspricht dem Stand von 6/88.

Die Ausarbeitung erfolgte durch ein Kollektiv des VEB Robotron-Elektronik Dresden Stammbetrieb des VEB Kombinat Robotron.

Herausgeber:

VEB Robotron-Vertrieb Berlin
Mohrenstr. 62
Berlin 1086

Inhaltsverzeichnis Seite

1. Einführung

Im Diagnosehandbuch Peripherie sind, beginnend ab Kapitel 2, alle Beschreibungen der Diagnoseprogramme enthalten, die für den Test der Peripherie des RVS K 1840 sowohl im Stand-alone-Modus als auch im User-Modus zur Verfügung stehen. Alle Diagnoseprogramme arbeiten unter der Steuerung des Diagnose-Supervisors PSSAA.
Die Beschreibungen sind in Abhängigkeit von der Peripheriekategorie in verschiedene Gruppen eingeteilt und erscheinen damit unter der der Peripheriekategorie zugeordneten Kapitelnummer. Werden durch ein Programm auch Geräte einer anderen Kategorie getestet, wird die Beschreibung nicht noch einmal aufgeführt (z.B. Test Seriendrucker mit Terminaldiagnoseprogramm PVTAA).
Nachfolgende Tabelle gibt einen Überblick, welches Gerät mit welchem Diagnoseprogramm getestet werden kann, und in welchem Kapitel die Beschreibung enthalten ist.

Gerätebezeichnung	physischer Name	Diagnoseprogramme	Kapitel
Terminal K 8911.80)	TG911)	PVTAA	2.1.
Terminal K 8911.81)	)	PVTBA	2.2.
IGT K 8918	TG918)		
Paralleldrucker VT23x00	DP23)	PVAAA	3.1.
Paralleldrucker VT270xx	DP27)		
Seriendrucker 1152/257	DT52)	PVTAA	2.1.
Seriendrucker K 6313)	DN13)	PVTBA	2.2.
Seriendrucker K 6314)	)		
HDG K 6404	GD04	PVTZA	4.1.
Plotter K 6411	GP11	PVTYA	4.2.
Magnetbandgerät CM 5306	MBG	PVMZA	5.1.
Wechselplattenspeicher CM 5404	PP05)	PVRAA	6.1.
Wechselplattenspeicher CM 5416	PP06)	PVRAC	6.2.
		PVRAD	6.3.
		PVRCA	6.5.
Festplattenspeicher K 5880	PM80	PVRAA	6.1.
		PVRAD	6.3.
		PVRDA	6.6.
		PVRGA	6.7.
		PVRGB	6.8.

2. Terminaldiagnoseprogramme

2.1. PVTAA: Lokalterminals Test A

2.1.1. Zusammenfassung

PVTAA ist ein Diagnoseprogramm der Ebene 2R (User-Modus,Dok.-Nr. 1.57.700609.7), das unter SVP 1800 läuft und den Terminal-Driver benötigt.
Das Programm führt den funktionellen Test von Lokalterminals. (Bildschirmterminals im VT100-Modus und Drucker) durch. Es beinhaltet 26 verschiedene Tests, wobei verschiedene Tests nur mit bestimmten Gerätetypen durchgeführt werden. Einige Tests verlangen Bedienerhandlungen. Zur Fehlererkennung ist bei den meisten Tests ein erzeugtes Muster visuell zu beurteilen.
Die Testreihenfolge wird durch den Bediener durch Auswahl einer der Sektionen QUALIFICATION, FUNCTIONAL, MANUAL und DEFAULT bestimmt.
Zum Festlegen der am zu testenden Gerät durchführbaren Tests wird vom Programm vor Testbeginn mit einer Steuerfolge der Gerätetyp abgefragt.
In den meisten Tests erfolgt durch Senden von ASCII-Kodes, -Steuerzeichen oder Steuerfolgen (ESC-Folgen) das Austesten der verschiedenen Terminalfunktionen. In 2 Echotests werden die Kodes von betätigten Tasten der Tastatur geechot. Ein spezieller Test gestattet bei Anschluß eines Kurzschlußsteckers an das Interfacekabel einen Interfacetest.
Die Durchführung von Tests an mehreren selektierten Geräten ist möglich, wobei erst alle spezifizierten Tests an einer Einheit durchgeführt werden, bevor der Test der nächsten Einheit beginnt.
Vor dem Start des Testprogrammes können mit dem Betriebssystemkommando SET TERMINAL... verschiedene Terminalcharakteristiken modifiziert werden.

2.1.2. Testbedingungen

2.1.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- Anschlußsteuerung AMF18 (asynchrone Kanäle)
- Bildschirmterminals:
  Terminal K 8911.80 oder K 8911.81 (TG911)
  Interaktives grafisches Terminal K 8918 (TG918)
  im Zeichenmodus
- Drucker:
  Typenraddrucker 1152/257 (DT52) mit ISO-Befehlssatz
  Nadeldrucker K 6313 oder K 6314 (DN13)
  mit ISO-Befehlssatz
- für Test 11: Kurzschlußstecker für Interfacekabel V.24

Es kann jedes andere Terminal als eines der Basisgeräte K 8911.80 (als Bildschirmterminal mit VT100-Verhalten) oder K 6313 (als Drucker mit ISO-Steuerzeichensatz) in Abhängigkeit von der Unterstützung durch das Betriebssystem getestet werden.

2.1.2.2. Software

- SVP 1800 mit Terminaldriver
- Diagnose-Supervisor

Es werden die Privilegien PHY_IO, LOG_IO und DIAGNOSE benötigt.

2.1.3. Voraussetzungen

- getestete Hardware
- getestete Anschlußsteuerung AMF18

2.1.4. Bedienhinweise

Nach dem Anlauf des SVP 1800 loggt sich der Bediener in das Betriebssystem ein und startet den Diagnose-Supervisor mit

$ RUN PSSAA

Vor dem Anschließen und Selektieren des zu testenden Gerätes ist das Testprogramm zu laden:

DS> LOAD PVTAA

Nun ist das zu testende Gerät mit Hilfe der Diagnose-Supervisorkommandos ATTACH und SELECT anzuschließen. Das kann im PROMT- oder im EXPLICIT-Modus erfolgen. Beispiel für das Anschließen eines Terminals K 8911.80 (TG911) über einen asynchronen Kanal der AMF18 an den SKRBUS-Adapter:

1. PROMT-Modus

```
DS> ATTACH                         ;Anschluß des SKRBUS-Adapters
DEVICE TYPE? DWA40
DEVICE LINK? HUB
DEVICE NAME? DW0
TR? 3
BR? 4
DS> ATT                            ;Anschluß des AMF18-Asynchronports
DEVICE TYPE? AMF18A
DEVICE LINK? DW0
DEVICE NAME? TXA
CSR? 760400
VECTOR? 310
BR? 5
ACTIVE LINES? 377
BAUD RATE? 9600
LOOPBACK TYPE? INTERNAL
SKRBUS INIT JUMPER? YES
DS> ATT                            ;Anschluß des Terminals K 8911.80
DEVICE TYPE? TG911
DEVICE LINK? TXA
DEVICE NAME? TXA2
```

2. EXPLICIT-Modus

```
DS> ATT DWA40 HUB DW0 3 4
DS> ATT AMF18A DW0 TXA 760400 310 5 377 9600 INTERNAL YES
DS> ATT TG911 TXA TXA2
```

Nach dem Selektieren kann des Testprogramm gestartet werden:

```
DS> SELECT TXA2                 ;Selektieren des Terminals TXA2
DS> START/switches              ;Start (Schalter s. Diagnose-Supervisor)
```

Das Testprogramm meldet sich mit folgender Ausschrift und arbeitet die spezifizierten Tests ab:

```
..PROGRAMM: PVTAA.....................     26 TESTS
```

Bei verschiedenen Tests werden Aufforderungen ausgegeben, die Bedienereingriffe erfordern oder es werden Reaktionen erläutert, die visuell zu verfolgen sind.

2.1.5. Funktionsbeschreibung

2.1.5.1. Übersicht

Lokalterminals unterschiedlicher Gerätetypen (Bildschirmterminals, Drucker) können durch das Testprogramm in den einzelnen Tests mit verschiedenen ASCII-Kodierungen, ASCII-Steuerzeichen und Steuerfolgen (ESC-Folgen) angesprochen werden. Das Verhalten des Terminals darauf wird getestet. Dies erfolgt durch visuelles Beurteilen der erzeugten Muster durch den Bediener. In verschiedenen Tests werden Bedienereingaben und Statusantworten des Terminals durch das Testprogramm analysiert und geechot. In einem Test erfolgt bei Anschluß eines Kurzschlußsteckers an das Interfacekabel ein Interfacetest.

2.1.5.2. Programmgröße

CC00 byte (hex.)
52224 byte (dez.)

2.1.5.3. Programmlaufzeit

Die Programmlaufzeit ist abhängig von der eingestellten Übertragungsgeschwindigkeit und vom Gerätetyp.
Die folgenden Programmlaufzeiten sind Richtwerte, gemessen bei einer Übertragungsgeschwindigkeit (Senden und Empfangen) von 9600 baud am Terminal K 8911.80 (TG911):

```
DEFAULT-Sektion        ca. 4 min
QUALIFICATION-Sektion  ca. 0,5 min
QUICK VERIFY-Modus     ca. 1,5 min
```

2.1.5.4. Abarbeitungsbedingungen

entfällt;

2.1.5.5. Flag-Verwendung

entfällt;

2.1.5.6. Allgemeine Programmanforderungen

Bildschirmterminals mit einem SETUP-Modus gestatten die Anzeige und das Ändern der Einstelldaten des Terminals. Durch das Betriebssystemkommando SHOW TERMINAL werden die im Betriebssystem eingestellten Daten angezeigt. Bezüglich der Übertragungsgeschwindigkeit, bit/Zeichen, Paritätskontrolle und Stoppbitanzahl ist prinzipiell Übereinstimmung in den eingestellten Daten zwischen Betriebssystem und Terminal herzustellen. Das kann durch das Betriebssystemkommando SET TERMINAL... oder im Terminal mit Hilfe des SETUP-Modus erfolgen.

2.1.5.7. Fehlermitteilung

entfällt;

2.1.5.8. Programmitteilungen

entfällt;

2.1.5.9. Leistungsparameter des Testobjektes

entfällt;

2.1.5.10. Aktionen bei unerwarteten Fehlern

entfällt;

2.1.5.11. Testfolge bei mehreren Einheiten

Sind mehrere Einheiten ausgewählt (Diagnose-Supervisor-Kommando SELECT), werden alle spezifizierten Tests erst an einer Einheit durchgeführt, bevor der Test der nächsten Einheit beginnt. Die Reihenfolge der Einheiten wird durch die Reihenfolge beim Auswählen festgelegt.

2.1.5.12. Einstellungen unter AFT

entfällt;

2.1.5.13. Testsektionen

PVTAA enthält die Sektionen DEFAULT, FUNCTIONAL, MANUAL und QUALIFICATION.

QUALIFICATION-Sektion
In der Qualifikationssektion wird ein prinzipieller Funktionstest durchgeführt (Datenweg, Zeichensatz). Folgende Tests gehören dazu:
Test-Nr. 1, 2, 6, 26

FUNCTIONAL-Sektion
In der Funktionalsektion werden alle wesentlichen Funktionen überprüft. In Abhängigkeit vom Gerätetyp können das folgende Tests sein:
Test-Nr. 1, ... 8, 12, ... 19, 21, 22, 25, 26

MANUAL-Sektion
In der Manuellsektion sind alle Tests vereint, die Bedienerhandlungen erfordern. In Abhängigkeit vom Gerätetyp können das folgende Tests sein:
Test-Nr. 9, 10, 11, 20, 23

DEFAULT-Sektion
Die Tests in der Standardsektion entsprechen denen in der FUNCTIONAL-Sektion.

Im QUICK VERIFY-Modus (d.h. das QUICK-Flag ist gesetzt) laufen in Abhängigkeit vom Getätetyp folgende Tests:
Test-Nr. 1, 2, 3, 6, 7, 8, 12, 17, 18, 26

2.1.5.14. Verschiedenes

- Da einige Tests nur auf bestimmten Terminaltypen laufen, wird vom Testprogramm der Gerätetyp durch Abfrage vor Testbeginn festgestellt. Das erfolgt mit Hilfe der "DEVICE ATTRIBUTES" ESC-Folge.
- Die Zeilenlänge der Terminals wird aus den SYSGEN-Daten entnommen.
- Für die Drucker sind nur die Tests-Nr. 1 bis 8 sinnvoll.

2.1.6. Testbeschreibungen

2.1.6.1. Test 1: DATA PATH
Datenweg

Der Datenweg über das Interface zum Terminal wird überprüft. Folgende 4 Zeilen werden ausgegeben:

```
*U*U*U*U*U*U*U*U*U*U...
U*U*U*U*U*U*U*U*U*U*...
@?@?@?@?@?@?@?@?@?@?...
?@?@?@?@?@?@?@?@?@?@...
```

Sie erzeugen auf dem Datenweg alternierend 1 und 0, gefolgt von 8 mal 1 und 8 mal 0. Die Reihenfolge wird 3 mal wiederholt. Im QUICK VERIFY-Modus werden nur 2 Zeilen ausgeführt.

:.1.6.2. Test 2: PRINTABLE CHARACTERS
Druckbare Zeichen

er Test erzeugt alle 95 druckbaren ASCII-Kodes (40...176 oktal). Jeder Kode ird 4 mal ausgegeben. Eine Vierergruppe ist durch 2 Leerzeichen getrennt. Die eilen sind auf die Terminalzeilenlänge zugeschnitten.

3eispiel:

```
''''  """"  ####  $$$$  %%%%  &&&&  ''''  ((((  ))))  ****
....  ////  0000  1111  2222  3333  4444  5555  6666  7777
;;;;  <<<<  ====  >>>>  ????  §§§§  AAAA  BBBB  CCCC  DDDD
:
:
```

2.1.6.3. Test 3: NON-PRINTABLE CHARACTERS
Nichtdruckbare Zeichen

ılle Kodes, die nicht druckbar sind und keine Steuerfunktion erzeugen, werden geprüft. Ein Ausgabepuffer, bestehend aus der Mnemonik des Kodes (7 byte) und n nal dem Kode (oktal) selbst, wobei n gleich der Zeilenlänge -7 ist, wird gleichzeitig gesendet. Der Bediener kann die angezeigte Mnemonik sehen, aber keine weiteren Reaktionen des Terminals.

Beispiel:

```
NUL 000
SOH 001
ACK 006
 .
 .
```

2.1.6.4. Test 4: CARRIAGE RETURN
Wagenrücklauf

Drucker:
Das Testprogramm sendet eine Zeile alternierend 0 und Leerzeichen, gefolgt von einem <CR>. Dann wird ein Puffer aufgebaut und gesendet, der aus n Leerzeichen, X, <CR>, (n+2) Leerzeichen, X, <CR>, (n+4) Leerzeichen,... besteht. Das hat den Effekt, daß die Leerzeichen der zuerst geschriebenen Zeile nacheinander durch X ersetzt werden:

```
zuerst: 0 0 0 0 0 0 0 0 0...
danach: 0X0X0X0X0X0X0X0X0...
```

Der Puffer kann für spezielle Terminals mit NUL gefüllt werden. Die Zeilenlänge entspricht der Zeilenlänge des Terminals. Der Bediener testet auf korrekte Aufreihung der Zeichen.

Bildschirmterminals:
Eine Zeile Sterne wird ausgegeben, gefolgt von einem <CR>. Die Anzahl der Sterne wird dann um 1 veringert und erneut eine Zeile ausgegeben usw. Es muß folgendes Muster entstehen:



2.1.6.5. Test 5: LINE FEED
Zeilenvorschub

Eine Referenzzeile wird ausgegeben (012345678901234...). Dann wird ein Puffer, bestehend aus (\, <LF>) (\, <LF>) (\, <LF>)... zum Terminal gesendet. Die Anzahl der (\, <LF>) ergibt sich durch die Anzahl von Zeilen, die notwendig sind zum Füllen des Bildschirmes (12 bei Drucker). Danach folgt wieder eine Referenzzeile und ein Puffer wird mit (n Leerzeichen) (\, <LF>) (\, <LF>)... gesendet, wobei n die Anzahl der Spalten ist, in denen bereits ein \ eingetragen wurde. Es muß sich folgendes Bild ergeben:

```
0123456789012345...
\
 \
  \
   \
```

2.1.6.6. Test 6: SPACE/ BACKSPACE
Leerzeichen/Rückleerzeichen

Das Testprogramm gibt eine Zeile von alternierenden / und Leerzeichen für Drukker oder alternierenden / und . für Bildschirmterminals aus. Dann wird durch Aussenden von (<BS>, <BS>, \) über die Zeile das / mit \ überdruckt. Als Resultat entsteht eine Zeile von alternierenden X und Leerzeichen bei Druckern bzw. alternierenden \ und . bei Bildschirmterminals. Pro Durchlauf werden 2 Zeilen erzeugt.

Drucker:

```
X X X X X X X X X X ...
X X X X X X X X X X ...
```

Bildschirmterminals:

```
\.\.\.\.\.\.\.\.\.\.\.\...
\.\.\.\.\.\.\.\.\.\.\.\...
```

2.1.6.7. Test 7: BELL
Signalton

Der Test sendet 5 BELL-Kodes (oktal 007). Er schreibt den Text BELL und sendet den Kode. Das akustische Signal muß zu hören sein.

2.1.6.8. Test 8: TAB
Tabulator

Der Test setzt einen gesetzten Tabulator aller 8 Spalten im Terminal voraus. Eine Referenzzeile wird ausgegeben (012345678901234...) und bei jedem gesetzten Tabulator ein U geschrieben (U, <TAB>, U, <TAB>,...). Danach wird ein Puffer aufgebaut und gesendet, der (U, nX, <TAB>, U, nX, <TAB>,...) enthält. 6 Zeilen werden ausgegeben, wobei n von 1 bis 6 läuft. Im QUICK VERIFY-Modus wird nur die Referenzzeile und eine TAB-Zeile ausgeführt.

```
01234567890123456789012345678...
U       U       U       U
UX      UX      UX      UX
UXX     UXX     UXX     UXX
UXXX    UXXX    UXXX    UXXX
```

2.1.6.9. Test 9: PATTERN ECHO
Muster-Echo

Der Test beginnt mit der Eingabeanforderung "ENTER PATTERN END WITH DEL KEY". Bis zu 132 beliebige Tasten können betätigt werden, wobei die DEL-Taste als Eingabeabschluß dient. Der Puffer wird dann exakt wieder ausgegeben. Kodes, die Steuerfunktionen entsprechen, müssen diese Funktionen auslösen. Das Testprogramm echot den Puffer, bis irgend eine Taste betätigt wird. Danach erfolgt wieder die Eingabeanforderung. Es kann eine neue Tastenfolge eingegeben werden. Mit sofortiger Eingabe von DEL wird der Test beendet.

2.1.6.10. Test 10: CHARACTER CODE ECHO
Zeichenkode-Echo

Nach der Eingabeanforderung "TYPE KEY FOLLOWED BY DEL" kann jede Taste (auch in Verbindung mit CONTROL oder SHIFT) betätigt werden. Der oktale Kode und das Zeichen, das dem Kode entspricht, werden verglichen. Der Bediener vergleicht den Kode mit der gedrückten Taste. Mit der DEL-Taste kann der Test beendet werden.

Beispiel:

Eingabe	Ausgabe
A	101 A
CTRL G	007 BEL ^G
TAB	011 HT ^I

2.1.6.11. Test 11: TURNAROUND TEST
Kurzschlußtest

Dieser Test ist nur für das Interface V.24 durchführbar.
Der Bediener wird gefragt, ob der Kurzschlußstecker an die zu testende Interfaceleitung angeschlossen wurde:

PUT TURNAROUND CONNECTOR ON ...
CONNECTOR ON (Y/N)

Das Testprogramm stellt dann den PASSALL-, NOECHO- und EIGHTBIT-Modus für das zu testende Terminal ein. Ein 24-byte-Puffer (mit gleitender 1, alternierendem 1 und 0, alles 1 und 0) wird ausgegeben. Die über den Kurzschluß empfangenen Daten werden mit den gesendeten verglichen. Treten keine Fehler auf, werden die eingestellten Modi zurückgesetzt. Der Kurzschlußstecker ist zu entfernen. Der Test ist beendet, wenn die Frage "TURNAROUND CONNECTOR OFF" mit YES beantwortet wird.

Bemerkung: Dieser Test läuft nicht auf dem Terminal, das der Bediener benutzt.

2.1.6.12. Test 12: CURSOR HOME Kursor Rücksetzen

Der Test löscht den Bildschirm und bewegt den Cursor unter Nutzung von <CR> und <LF> auf die HOME-Position. Die folgende Nachricht wird dann ab Zeile 1, Spalte 1 ausgegben:

X <--X OVERWRITTEN BY 1
CURSOR HOME TEST

Nach kurzer Zeit wird dann eine "CURSOR HOME" ESC-Folge gesendet, gefolgt von einer 1. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

1 <--X OVERWRITTEN BY 1

2.1.6.13. Test 13: CURSOR UP Kursor nach oben

Der Test löscht den Bildschirm unter Nutzung von <LF>. Die folgende Nachricht wird ab Zeile 1, Spalte 1 ausgegeben:

CURSOR UP TEST
X OVERWRITTEN BY 1--> X

Der Cursor wird dann 10 Zeilen abwärts bewegt und eine 2 ausgegeben. Nach kurzer Zeit wird BACKSPACE, 10 mal die "CURSOR UP" ESC-Folge und eine 1 gesendet. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

X OVERWRITTEN BY 1--> 1

2.1.6.14. Test 14: CURSOR RIGHT Kursor nach Rechts

Der Test löscht den Bildschirm unter Nutzung von <LF>. Der Cursor wird auf die HOME-Position bewegt und die folgende Nachricht angezeigt:

CURSOR RIGHT TEST
X OVERWRITTEN BY 1--> X

Der Cursor wird wieder an den linken Rand bewegt. Unter Nutzung von "CURSOR

RIGHT" ESC-Folgen wird der Cursor dann nach rechts zur Position des X bewegt und eine 1 ausgegeben. Es entsteht folgendes Bild:

```
X OVERWRITTEN BY 1--> 1
```

2.1.6.15. Test 15: LINE ERASE Zeile löschen

Der Test füllt den Bildschirm mit E und bewegt dann den Cursor auf die HOME-Position. Unter Nutzung der "CURSOR RIGHT" und "ERASE IN LINE" ESC-Folgen wird in jeder Zeile ein Teil der E gelöscht, so daß ein Stufenmuster entsteht.

Beispiel:

```
EEE
EEEEEE
EEEEEEEEE
EEEEEEEEEEEE
EEEEEEEEEEEEEEE
  .
  .
```

Für ANSI-kompatible Terminals werden 2 zusätzliche Tests durchgeführt:
Der 1. Test überprüft die Möglichkeit des Löschens vom Zeilenanfang. Der Bildschirm wird zuerst mit E gefüllt, dann wird unter Nutzung der "CURSOR RIGHT" und "ERASE FROM START OF LINE" ESC-Folgen ein negatives Stufenmuster erzeugt.

Beispiel:

```
EEEEEEEEEEEEEEEEEEEEE...
   EEEEEEEEEEEEEEEEEE...
      EEEEEEEEEEEEEEE...
         EEEEEEEEEEEE...
            EEEEEEEEE...
```

Der 2. Test füllt den Bildschirm mit E und unter Nutzung der "CURSOR DOWN" und "ERASE ENTIRE LINE" ESC-Folgen wird jede 2. Zeile gelöscht.

Beispiel:

```
EEEEEEEEEEEEEEEEEEEEEE...

EEEEEEEEEEEEEEEEEEEEEE...

EEEEEEEEEEEEEEEEEEEEEE...
  .
  .
```

2.1.6.16. Test 16: SCREEN ERASE
Bildschirm löschen

Teil 1:
Nach der Überschrift "SCREEN ERASE 16" wird der Bildschirm mit E gefüllt. Dann wird der Cursor in die Mitte der untersten Zeile bewegt. Das Testprogramm sendet danach die "ERASE FROM CURSOR TO END OF SCREEN" ESC-Folge. Danach wird der Cursor eine Zeile nach oben bewegt. Diese Testfolge wird wiederholt, bis der Cursor auf der obersten Zeile steht und eine halbe Zeile E auf dem Bildschirm stehen geblieben ist.

Teil 2: (nur für ANSI-Terminals)
Nach der Überschrift "ERASE FROM START OF SCREEN TO CURSOR" wird der Bildschirm mit E gefüllt und der Cursor in die Mitte der obersten Zeile bewegt. Durch Senden der "ERASE FROM BEGINNING OF SCREEN TO CURSOR" ESC-Folge wird der Bildschirm vom Bildschirmanfang bis zum Cursor gelöscht. Dann wird der Cursor eine Zeile abwärts bewegt. Diese Testfolge wird solange wiederholt, bis der Cursor auf der untersten Zeile steht und eine halbe Zeile E am Bildschirmende stehen bleibt.
Nach der Überschrift "ERASE ENTIRE SCREEN" wird der Bildschirm nochmals mit E gefüllt und durch Senden der "ERASE ENTIRE SCREEN" ESC-Folge vollständig gelöscht.

2.1.6.17. Test 17: CURSOR LEFT
Kursor nach links

Der Test löscht den Bildschirm unter Nutzung von <LF>. Der Cursor wird auf die HOME-Position bewegt und die folgende Nachricht ausgegeben:

```
CURSOR LEFT TEST
X <--X OVERWRITTEN BY 1
```

Der Cursor steht danach am Ende der Nachricht. Durch Nutzung der "CURSOR LEFT" ESC-Folge wird der Cursor über das X positioniert und eine 1 ausgegeben. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

```
1 <--X OVERWRITTEN BY 1
```

2.1.6.18. Test 18: CURSOR DOWN
Kursor nach unten

Zuerst wird der Bildschirm gelöscht und die folgende Nachricht ausgegeben:

```
CURSOR DOWN TEST - CREATES A DIAGONAL LINE
```

Der Cursor wird an den linken Rand bewegt. Unter Nutzung der "CURSOR DOWN" ESC-Folge wird der Cursor eine Zeile nach unten bewegt und ein \ ausgegeben. Das wird 10 mal wiederholt, so daß eine diagonale Linie von oben links nach unten rechts erzeugt wird.

Beispiel:



2.1.6.19. Test 19: DIRECT CURSOR ADDRESSING Direkte Kursoradressierung

Der Test erzeugt ein Spiralmuster von 104 Sternen. Dieses ist am Bildschirm durch Begutachtung jeder Zeichenposition, die durch die "DIRECT CURSOR ADDRESSING" ESC-Folge festgelegt wurde, zu überprüfen. Der Bildschirm muß folgendes Muster anzeigen:

```
CURSOR ADDRESSING TEST
*************
            *
*********** *
*         * *
* ******* * *
* *     * * *
* * *** * * *
* * * * * * *
* * *   * * *
* * ***** * *
* *       * *
* ********* *
*           *
*************
```

2.1.6.20. Test 20: AUXILLARY KEYPAD Zusatztastenfeld

Wird die Frage "ALTERNATE KEYPAD? (Y/N)" mit YES beantwortet, setzt das Testprogramm das Terminal in den KEYPAD- und CURSOR KEY-MODUS. Der Bediener wird dann mit der Nachricht

TYPE ANY KEY ON AUXILIARY KEYPAD THEN TYPE DEL KEY, DEL ONLY TO EXIT

aufgefordert, Tasten des Zusatztastenfeldes und die Cursortaste zu betätigen. Durch Drücken der DEL-Taste wird der Test abgebrochen. Alle gedrückten Tasten werden auf die Ausgabe des korrekten Kodes getestet. Der empfangene Kode wird als Tastensymbol oder als Mnemonik ausgegeben.

2.1.6.21. Test 21: REVERSE LINEFEED / INDEX Inverser Zeilenvorschub

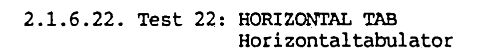
Der Bildschirm wird gelöscht und der Cursor auf die HOME-Position bewegt. Auf der obersten Zeile wird der Testname ausgegeben und danach der Cursor auf die

unterste Zeile, Spalte 1, bewegt. Dann wird unter Nutzung von "REVERSE LINEFEED" oder "REVERSE INDEX" ESC-Folgen und Ausgeben eines / eine diagonale Linie von unten links nach oben rechts erzeugt.

Beispiel:

```
  REVERSE LINEFEED - INDEX TEST CREATES A DIAGONAL LINE
                                     /
                                    /
                                   /
                                  /
                                 /
```

2.1.6.22. Test 22: HORIZONTAL TAB
Horizontaltabulator



Es wird die Fähigkeit des Terminals zum Setzen und Löschen von Tabulatoren unter Steuerung eines Hostrechners getestet. Der Test besteht aus 2 Teilen.

Teil 1: (Für mechanische Tabulatoren)
Der Bediener wird gefragt, ob die Option vorhanden ist. Wenn YES, wird über die Seite eine Referenzlinie erzeugt, die durch Tabulatorstopps mit 1, 2, 3, ...15 Spalten Abstand zerlegt wird. Drei Zeilen, bestehend aus TAB, I, TAB, I, ..., werden unter die Referenzzeile gedruckt. Die I müssen in einer Linie unter den V in der Referenzzeile stehen.

Beispiel:

```
  .V..V...V....V.....V......V...
   I  I   I    I     I      I...
   I  I   I    I     I      I...
   I  I   I    I     I      I...
```

Teil 2: (Für Terminals, die durch ESC-Folgen gesteuert werden)
Eine Refenzzeile, bestehend aus . und V wird über die Seite gedruckt. Dann werden Tabulatorstopps korrespondierend mit den V in der Refenzzeile gesetzt. Drei Zeilen, bestehend aus TAB, I, TAB, I, ... werden dann unter die Refenzzeile gedruckt. Die I müssen in einer Linie unter den V stehen. Dieser Test wird für Tabulatorabstände von 2, 4, 8, 16 und 32 durchgeführt.

Beispiel für Tabulatorabstand 4:

```
  ...V...V...V...V...V...
     I   I   I   I   I
     I   I   I   I   I
     I   I   I   I   I
```

2.1.6.23. Test 23: ANSWERBACK
Rückantwort

Wird die Frage "ANSWERBACK / HERE IS? (Y/N)" mit YES beantwortet, sendet der Test einen ENQ-Kode zum Terminal. Dann liest er die "HERE IS" Nachricht und sendet sie zu der zu testenden Einheit zurück:

```
SENT ENQ (005) RECEIVED MESSAGE
.............                                  (empfangene Antwort)
```

Der Bediener wird dann aufgefordert, die "ANSWERBACK" Taste zu drücken. Die Nachricht wird gelesen und zurückgesendet:

```
PRESS HERE IS KEY, MESSAGE IS
.............                                  (empfangene Antwort)
```

Die Nachrichten werden nicht auf ihren Inhalt getestet. Der Test überprüft, daß beide Male dieselbe Nachricht übertragen wird. Mit <CR> kann der Test beendet werden. Eingabe eines Zeichens und <CR> wiederholt den Test.

Bemerkung: Beim Terminal K 8911.80 besitzt die Taste PF12 die "ANSWERBACK"-Funktion. Diese ist vor Testbeginn entsprechend der Bedienungsanleitung zu programmieren. Nach der Aufforderung "PRESS HERE IS KEY" ist CTRL/PF12 zu drücken.

2.1.6.24. Test 24: reserviert

2.1.6.25. Test 25: GRAPHICS CHARACTER SET Grafikzeichensatz

Der Test versetzt das Terminal in den Grafikmodus und schreibt nach Ausgabe der Überschrift:

```
GRAPHICS CHARACTER SET TEST
```

10 Zeilen mit Zeichen des Grafikzeichensatzes der Kodes 137 bis 176 oktal. Der Grafiksatz ist für verschiedene Terminals unterschiedlich.

2.1.6.26. Test 26: SCROLLING Bildschirmrollen

Der Test prüft die Fähigkeit des Terminals, korrekt in den beiden Modi JUMP und SMOOTH SCROLLING zu arbeiten. Der Test füllt 22 Zeilen mit . als Hintergrund. Dann wird ein Rollbereich von Zeile 4 bis 12 definiert. 36 Zeilen eines Musters werden dann gesendet, die nur innerhalb des Rollbereiches erscheinen dürfen und nach oben rollen, wenn der Rollbereich gefüllt ist. Das Muster wird nochmals gesendet unter Nutzung der "REVERSE INDEX" ESC-Folge anstelle von <LF>. In diesem Falle rollt der Bereich nach unten, wenn der Rollbereich gefüllt ist.

Beispiel:

```
SCROLLING TEST ......................
JUMP MODE ...........................
.....................................
.....................................
         ABCDEFGHIJKL
         BCDEFGHIJKLM
         CDEFGHIJKLMN
         DEFGHIJKLMNO
         EFGHIJKLMNOP
.....................................
.....................................
```

Nach dem Test des JUMP-Modus erfolgt der gesamte Testablauf noch einmal für den SMOOTH-Modus.

2.1.7. Auflistung der Tests

Test 1: DATA PATH
Datenweg
Test 2: PRINTABLE CHARACTERS
Druckbare Zeichen
Test 3: NON-PRINTABLE CHARACTERS
Nichtdruckbare Zeichen
Test 4: CARRIAGE RETURN
Wagenrücklauf
Test 5: LINE FEED
Zeilenvorschub
Test 6: SPACE / BACKSPACE
Leerzeichen/Rückleerzeichen
Test 7: BELL
Signalton
Test 8: TAB
Tabulator
Test 9: PATTERN ECHO
Muster-Echo
Test 10: CHARACTER CODE ECHO
Zeichenkode-Echo
Test 11: TURNAROUND TEST
Kurzschlußtest
Test 12: CURSOR HOME
Kursor Rücksetzen
Test 13: CURSOR UP
Kursor nach oben
Test 14: CURSOR RIGHT
Kursor nach rechts
Test 15: LINE ERASE
Zeile löschen
Test 16: SCREEN ERASE
Bildschirm löschen
Test 17: CURSOR LEFT
Kursor nach links
Test 18: CURSOR DOWN
Kursor nach unten
Test 19: DIRECT CURSOR ADDRESSING
Direkte Kursoradressierung
Test 20: AUXILLARY KEYPAD
Zusatztastenfeld
Test 21: REVERSE LINEFEED / INDEX
Inverser Zeilenvorschub
Test 22: HORIZONTAL TAB
Horizontaltabulator
Test 23: ANSWERBACK
Rückantwort
Test 24: reserviert
Test 25: GRAPHICS CHARACTER SET
Grafikzeichensatz
Test 26: SCROLLING
Bildschirmrollen

2.2. PVTBA: Lokalterminals Test B

2.2.1. Zusammenfassung

PVTBA ist ein Diagnoseprogramm der Ebene 2R (User-Modus, Dok.-Nr. 1.57.700610.3), das unter SVP 1800 läuft und den Terminal-Driver benötigt.
Das Programm führt den funktionellen Test von Lokalterminals (Bildschirmterminals im VT100-Modus und Drucker) durch. Es beinhaltet 20 verschiedene Tests, wobei verschiedene Tests nur mit bestimmten Gerätetypen durchgeführt werden. Zur Fehlererkennung ist bei den meisten Tests ein erzeugtes Muster visuell zu beurteilen.
Die Testreihenfolge wird durch den Bediener durch Auswahl einer der Sektionen QUALIFICATION, FUNCTIONAL und DEFAULT bestimmt.
Zum Festlegen der am zu testenden Gerät durchführbaren Tests wird vom Programm vor Testbeginn mit einer Steuerfolge der Gerätetyp abgefragt.
In den Tests erfolgt durch Senden von ASCII-Kodes, ASCII-Steuerzeichen oder Steuerfolgen (ESC-Folgen) das Austesten der verschiedenen Terminalfunktionen.
Es können mehrere Geräte parallel getestet werden. Dabei wird ein Test erst an allen selektierten Geräten durchgeführt, bevor der nächste spezifizierte Test beginnt. Die Gerätereihenfolge entspricht der Reihenfolge beim Selektieren.
Vor dem Start des Testprogrammes können mit dem Betriebssystemkommando SET TERMINAL... verschiedene Terminalcharakteristiken modifiziert werden.

2.2.2. Testbedingungen

2.2.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- Anschlußsteuerung AMF18 (asynchrone Kanäle)
- Bildschirmterminals:
  Terminal K 8911.80 oder K 8911.81 (TG911)
  Interaktives grafisches Terminal K 8918 (TG918)
  im Zeichenmodus
- Drucker:
  Typenraddrucker 1152/257 (DT52) mit ISO-Befehlssatz
  Nadeldrucker K 6313 oder K 6314 (DN13)
  mit ISO-Befehlssatz

2.2.2.2. Software

- SVP 1800 mit Terminaldriver
- Diagnose-Supervisor

Es werden die Privilegien PHY_IO, LOG_IO und DIAGNOSE benötigt.

2.2.3. Voraussetzungen

- vollständig getesteter Rechnerkern
- getestete Anschlußsteuerung AMF18

2.2.4. Bedienhinweise

Nach dem Anlauf des SVP 1800 loggt sich der Bediener in das Betriebssystem ein und startet den Diagnose-Supervisor mit

$ RUN PSSAA

Vor dem Anschließen und Auswählen des zu testenden Gerätes ist das Testprogramm zu laden:

DS> LOAD PVTBA

Nun ist das zu testende Gerät mit Hilfe der Diagnose-Supervisorkommandos ATTACH und SELECT anzuschließen. Das kann im PROMT- oder im EXPLICIT-Modus erfolgen. Beispiel für das Anschließen eines Terminals K 8911.80 (TG911) über einen asynchronen Kanal der AMF18 an den SKRBUS-Adapter:

1. PROMT-Modus

```
DS> ATTACH                          ;Anschluß des SKRBUS-Adapters
DEVICE TYPE? DWA40
DEVICE LINK? HUB
DEVICE NAME? DW0
TR? 3
BR? 4
DS> ATT                             ;Anschluß des AMF18-Asynchronports
DEVICE TYPE? AMF18A
DEVICE LINK? DW0
DEVICE NAME? TXA
CSR? 760400
VECTOR? 310
BR? 5
ACTIVE LINES? 377
BAUD RATE? 9600
LOOPBACK TYPE? INTERNAL
SKRBUS INIT JUMPER? YES
DS> ATT                             ;Anschluß des Terminals K 8911.80
DEVICE TYPE? TG911
DEVICE LINK? TXA
DEVICE NAME? TXA2
```

2. EXPLICIT-Modus

```
DS> ATT DWA40 HUB DW0 3 4
DS> ATT AMF18A DW0 TXA 760400 310 5 377 9600 INTERNAL YES
DS> ATT TG911 TXA TXA2
```

Nach dem Auswählen kann des Testprogramm gestartet werden:

```
DS> SELECT TXA2             ;Auswählen des Terminals TXA2
DS> START/switches          ;Start (Schalter s. Diagnose-Supervisor)
```

Das Testprogramm meldet sich mit folgender Ausschrift und arbeitet die spezifizierten Tests ab:

```
..PROGRAMM: PVTBA....................     20 TESTS
```

2.2.5. Funktionsbeschreibung

2.2.5.1. Übersicht

Lokalterminals unterschiedlicher Gerätetypen (Bildschirmterminals, Drucker) können durch das Testprogramm in den einzelnen Tests mit verschiedenen ASCII-Kodierungen, ASCII-Steuerzeichen und Steuerfolgen (ESC-Folgen) angesprochen werden. Das Verhalten des Terminals darauf wird getestet. Dies erfolgt durch visuelles Beurteilen der erzeugten Muster durch den Bediener.

2.2.5.2. Programmgröße

AE00 byte (hex.)
44544 byte (dez.)

2.2.5.3. Programmlaufzeit

Die Programmlaufzeit ist abhängig von der eingestellten Übertragungsgeschwindigkeit und vom Gerätetyp.
Die folgenden Programmlaufzeiten sind Richtwerte, gemessen bei einer Übertragungsgeschwindigkeit (Senden und Empfangen) von 9600 baud am Terminal K 8911.80 (TG911):

DEFAULT-Sektion	ca. 4 min
QUALIFICATION-Sektion	ca. 0,5 min
QUICK VERIFY-Modus	ca. 1,5 min

2.2.5.4. Abarbeitungsbedingungen

entfällt

2.2.5.5. Flag-Verwendung

entfällt

2.2.5.6. Allgemeine Programmanforderungen

Bildschirmterminals mit einem SETUP-Modus gestatten die Anzeige und das Ändern der Einstelldaten des Terminals. Durch das Betriebssystemkommando SHOW TERMINAL werden die im Betriebssystem eingestellten Daten angezeigt. Bezüglich der Übertragungsgeschwindigkeit, bit/Zeichen, Paritätskontrolle und Stoppbitanzahl ist prinzipiell Übereinstimmung in den eingestellten Daten zwischen Betriebssystem und Terminal herzustellen. Das kann durch das Betriebssystemkommando SET TERMINAL... oder im Terminal mit Hilfe des SETUP-Modus erfolgen.

2.2.5.7. Fehlerermittlung

entfällt

2.2.5.8. Programmitteilungen

entfällt

2.2.5.9. Leistungsparameter des Testobjektes

entfällt

2.2.5.10. Aktionen bei unerwarteten Fehlern

entfällt

2.2.5.11. Testfolge bei mehreren Einheiten

PVTBA ist für den Test mehrerer Einheiten vorgesehen. Es wird ein Test erst an allen ausgewählten Einheiten durchgeführt, bevor der nächste spezifizierte Test beginnt. Die Reihenfolge der Einheiten wird durch die Reihenfolge beim Auswählen (Diagnose-Supervisor-Kommando SELECT) festgelegt.

2.2.5.12. Einstellungen unter AFT

entfällt

2.2.5.13. Testsektionen

PVTBA enthält die Sektionen DEFAULT, FUNCTIONAL und QUALIFICATION.

QUALIFICATION-Sektion
In der Qualifikationsektion wird ein prinzipieller Funktionstest durchgeführt (Datenweg, Zeichensatz). Folgende Tests gehören dazu:
Test-Nr. 1, 2, 6, 20

FUNCTIONAL-Sektion
In der Funktionalsektion werden alle wesentlichen Funktionen überprüft. In Abhängigkeit vom Gerätetyp können das alle vorhandenen Tests sein.

DEFAULT-Sektion
Die Tests in der Standardsektion entsprechen denen in der FUNCTIONAL-Sektion.

Im QUICK VERIFY-Modus (d.h. das QUICK-Flag ist gesetzt) laufen in Abhängigkeit vom Getätetyp folgende Tests:
Test-Nr. 1, 3, 6 ... 11, 14, 15, 20

2.2.5.14. Verschiedenes

- Da einige Tests nur auf bestimmten Terminaltypen laufen, wird vom Testprogramm der Gerätetyp durch Abfrage vor Testbeginn festgestellt. Das erfolgt mit Hilfe der "DEVICE ATTRIBUTES" ESC-Folge.
- Die Zeilenlänge der Terminals wird aus den SYSGEN-Daten entnommen.
- Für die Drucker sind nur die Tests-Nr. 1 bis 8 sinnvoll.

2.2.6. Testbeschreibungen

2.2.6.1. Test 1: DATA PATH
Datenweg

Der Datenweg über das Interface zum Terminal wird überprüft. Folgende 4 Zeilen werden ausgegeben:

```
*U*U*U*U*U*U*U*U*U*U...
U*U*U*U*U*U*U*U*U*U*...
@?@?@?@?@?@?@?@?@?@?...
?@?@?@?@?@?@?@?@?@?@...
```

Sie erzeugen auf dem Datenweg alternierend 1 und 0, gefolgt von 8 mal 1 und 8 mal 0. Die Reihenfolge wird 3 mal wiederholt. Im QUICK VERIFY-Modus werden nur 2 Zeilen ausgeführt.

2.2.6.2. Test 2: PRINTABLE CHARACTERS
Druckbare Zeichen

Der Test erzeugt alle 95 druckbaren ASCII-Kodes (40...176 oktal). Jeder Kode wird 4 mal ausgegeben. Eine Vierergruppe ist durch 2 Leerzeichen getrennt. Die Zeilen sind auf die Terminalzeilenlänge zugeschnitten.

Beispiel:

```
!!!!  """"  ####  SSSS  %%%%  &&&&  ''''  ((((  ))))  ++++
....  ////  0000  1111  2222  3333  4444  5555  6666  7777
;;;;  <<<<  ====  >>>>  ????  §§§§  AAAA  BBBB  CCCC  DDDD
.
.
```

2.2.6.3. Test 3: NON-PRINTABLE CHARACTERS
Nichtdruckbare Zeichen

Alle Kodes, die nicht druckbar sind und keine Steuerfunktion erzeugen, werden geprüft. Ein Ausgabepuffer, bestehend aus der Mnemonik des Kodes (7 byte) und n mal dem Kode (oktal) selbst, wobei n gleich der Zeilenlänge - 7 ist, wird gleichzeitig gesendet. Der Bediener kann die angezeigte Mnemonik sehen, aber keine weiteren Reaktionen des Terminals.

Beispiel:

```
NUL 000
SOH 001
ACK 006
 .
 .
```

2.2.6.4. Test 4: CARRIAGE RETURN Wagenrücklauf

Drucker:
Das Testprogramm sendet eine Zeile alternierend 0 und Leerzeichen, gefolgt von einem <CR>. Dann wird ein Puffer aufgebaut und gesendet, der aus n Leerzeichen, X, <CR>, (n+2) Leerzeichen, X, <CR>, (n+4) Leerzeichen,... besteht. Das hat den Effekt, daß die Leerzeichen der zuerst geschriebenen Zeile nacheinander durch X ersetzt werden:

```
zuerst: 0 0 0 0 0 0 0 0 0...
danach: 0X0X0X0X0X0X0X0X0...
```

Der Puffer kann für spezielle Terminals mit NUL gefüllt werden. Die Zeilenlänge entspricht der Zeilenlänge des Terminals. Der Bediener testet auf korrekte Aufreihung der Zeichen.

Bildschirmterminals:
Eine Zeile Sterne wird ausgegeben, gefolgt von einem <CR>. Die Anzahl der Sterne wird dann um 1 verringert und erneut eine Zeile ausgegeben usw. Es muß folgendes Muster entstehen:

```
*************************
************************
***********************
**********************
.
```

2.2.6.5. Test 5: LINEFEED Zeilenvorschub

Eine Referenzzeile wird ausgegeben (012345678901234...). Dann wird ein Puffer, bestehend aus (\, <LF>) (\, <LF>) (\, <LF>)... zum Terminal gesendet. Die Anzahl der (\, <LF>) ergibt sich durch die Anzahl von Zeilen, die zum Füllen des Bildschirmes (12 bei Drucker) notwendig sind. Danach folgt wieder eine Referenzzeile und ein Puffer mit (n Leerzeichen) (\, <LF>) (\, <LF>)... wird gesendet, wobei n die Anzahl der Spalten ist, in denen bereits ein \ eingetragen wurde. Es muß sich folgendes Bild ergeben:

```
0123456789012345...
\
 \
  \
   \
```

2.2.6.6. Test 6: SPACE / BACKSPACE
Leerzeichen/Rückleerzeichen

Das Testprogramm gibt eine Zeile von alternierenden / und Leerzeichen für Drukker oder alternierenden / und . für Bildschirmterminals aus. Dann wird durch Aussenden von (<BS>, <BS>, \) über die Zeile das / mit \ überdruckt. Als Resultat entsteht eine Zeile von alternierenden X und Leerzeichen bei Druckern bzw. alternierenden \ und . bei Bildschirmterminals. Pro Durchlauf werden 2 Zeilen erzeugt.

Drucker:

```
X X X X X X X X X X ...
X X X X X X X X X X ...
```

Bildschirmterminals:

```
\.\.\.\.\.\.\.\.\.\.\...
\.\.\.\.\.\.\.\.\.\.\...
```

2.2.6.7. Test 7: BELL
Signalton

Der Test sendet 5 BELL-Kodes (oktal 007). Er schreibt den Text BELL und sendet den Kode. Das akustische Signal muß zu hören sein.

2.2.6.8. Test 8: TAB
Tabulator

Der Test setzt einen gesetzten Tabulator aller 8 Spalten im Terminal voraus. Eine Referenzzeile wird ausgegeben (012345678901234...) und bei jedem gesetzten Tabulator ein U geschrieben (U, <TAB>, U, <TAB>,...). Danach wird ein Puffer aufgebaut und gesendet, der (U, nX, <TAB>, U, nX, <TAB>,...) enthält. 6 Zeilen werden ausgegeben, wobei n von 1 bis 6 läuft. Im QUICK VERIFY-Modus wird nur die Referenzzeile und eine TAB-Zeile ausgeführt.

```
012345678901234567890123456 78...
U       U       U       U
UX      UX      UX      UX
UXX     UXX     UXX     UXX
UXXX    UXXX    UXXX    UXXX
 .
```

2.2.6.9. Test 9: CURSOR HOME
Kursor Rücksetzen

Der Test löscht den Bildschirm und bewegt den Cursor unter Nutzung von <CR> und <LF> auf die HOME-Position. Die folgende Nachricht wird dann ab Zeile 1, Spalte 1 ausgegben:

```
X <—X OVERWRITTEN BY 1
CURSOR HOME TEST
```

Nach kurzer Zeit wird dann eine "CURSOR HOME" ESC-Folge gesendet, gefolgt von einer 1. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

```
1 <—X OVERWRITTEN BY 1
```

2.2.6.10. Test 10: CURSOR UP
Kursor nach oben

Der Test löscht den Bildschirm unter Nutzung von <LF>. Die folgende Nachricht wird ab Zeile 1, Spalte 1 ausgegeben:

```
CURSOR UP TEST
X OVERWRITTEN BY 1—> X
```

Der Cursor wird dann 10 Zeilen abwärts bewegt und eine 2 ausgegeben. Nach kurzer Zeit wird BACKSPACE, 10 mal die "CURSOR UP" ESC-Folge und eine 1 gesendet. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

```
X OVERWRITTEN BY 1—> 1
```

2.2.6.11. Test 11: CURSOR RIGHT
Kursor nach rechts

Der Test löscht den Bildschirm unter Nutzung von <LF>. Der Cursor wird auf die HOME-Position bewegt und die folgende Nachricht angezeigt:

```
CURSOR RIGHT TEST
X OVERWRITTEN BY 1—> X
```

Der Cursor wird wieder an den linken Rand bewegt. Unter Nutzung von "CURSOR RIGHT" ESC-Folgen wird der Cursor dann nach rechts zur Position des X bewegt und eine 1 ausgegeben. Es entsteht folgendes Bild:

```
X OVERWRITTEN BY 1—> 1
```

2.2.6.12. Test 12: LINE ERASE
Zeile löschen

Der Test füllt den Bildschirm mit E und bewegt dann den Cursor auf die HOME-Position. Unter Nutzung der "CURSOR RIGHT" und "ERASE IN LINE" ESC-Folgen wird in jeder Zeile ein Teil der E gelöscht, so daß ein Stufenmuster entsteht.

Beispiel:

```
EEE
EEEEEE
EEEEEEEEE
EEEEEEEEEEEE
EEEEEEEEEEEEEEE
 .
 .
```

Für ANSI-kompatible Terminals werden 2 zusätzliche Tests durchgeführt:
Der 1. Test überprüft die Möglichkeit des Löschens vom Zeilenanfang. Der Bildschirm wird zuerst mit E gefüllt, dann wird unter Nutzung der "CURSOR RIGHT" und "ERASE FROM START OF LINE" ESC-Folgen ein negatives Stufenmuster erzeugt.
Beispiel:

```
EEEEEEEEEEEEEEEEEEEEE...
   EEEEEEEEEEEEEEEEEE...
      EEEEEEEEEEEEEEE...
         EEEEEEEEEEEE...
            EEEEEEEEE...
```

Der 2. Test füllt den Bildschirm mit E und unter Nutzung der "CURSOR DOWN" und "ERASE ENTIRE LINE" ESC-Folgen wird jede 2. Zeile gelöscht.

Beispiel:

```
EEEEEEEEEEEEEEEEEEEEEEE...

EEEEEEEEEEEEEEEEEEEEEEE...

EEEEEEEEEEEEEEEEEEEEEEE...
 .
 .
```

2.2.6.13. Test 13: SCREEN ERASE Bildschirm löschen

Teil 1:
Nach der Überschrift "SCREEN ERASE" wird der Bildschirm mit E gefüllt. Dann wird der Cursor in die Mitte der untersten Zeile bewegt. Das Testprogramm sendet danach die "ERASE FROM CURSOR TO END OF SCREEN" ESC-Folge. Danach wird der Cursor eine Zeile nach oben bewegt. Diese Testfolge wird wiederholt, bis der Cursor auf der obersten Zeile steht und eine halbe Zeile E auf dem Bildschirm stehen geblieben ist.

Teil 2: (nur für ANSI-Terminals)
Nach der Überschrift "ERASE FROM START OF SCREEN TO CURSOR" wird der Bildschirm mit E gefüllt und der Cursor in die Mitte der obersten Zeile bewegt. Durch Senden der "ERASE FROM BEGINNING OF SCREEN TO CURSOR" ESC-Folge wird der Bildschirm vom Bildschirmanfang bis zum Cursor gelöscht. Dann wird der Cursor eine Zeile abwärts bewegt. Diese Testfolge wird solange wiederholt, bis der Cursor auf der untersten Zeile steht und eine halbe Zeile E am Bildschirmende stehen bleibt.
Nach der Überschrift "ERASE ENTIRE SCREEN" wird der Bildschirm nochmals mit E

gefüllt und durch Senden der "ERASE ENTIRE SCREEN" ESC-Folge vollständig gelöscht.

2.2.6.14. Test 14: CURSOR LEFT
Kursor nach links

Der Test löscht den Bildschirm unter Nutzung von <LF>. Der Cursor wird auf die HOME-Position bewegt und die folgende Nachricht ausgegeben:

CURSOR LEFT TEST
X <--X OVERWRITTEN BY 1

Der Cursor steht danach am Ende der Nachricht. Durch Nutzung der "CURSOR LEFT" ESC-Folge wird der Cursor über das X positioniert und eine 1 ausgegeben. Die 1 muß das X überschreiben. Es entsteht folgendes Bild:

1 <--X OVERWRITTEN BY 1

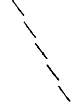
2.2.6.15. Test 15: CURSOR DOWN
Kursor nach unten

Zuerst wird der Bildschirm gelöscht und die folgende Nachricht ausgegeben:

CURSOR DOWN TEST - CREATES A DIAGONAL LINE

Der Cursor wird an den linken Rand bewegt. Unter Nutzung der "CURSOR DOWN" ESC-Folge wird der Cursor eine Zeile nach unten bewegt und ein \ ausgegeben. Das wird 10 mal wiederholt, so daß eine diagonale Linie von oben links nach unten rechts erzeugt wird.

Beispiel:



2.2.6.16. Test 16: DIRECT CURSOR ADDRESSING
Direkte Kursoradressierung

Der Test erzeugt ein Spiralmuster von 104 Sternen. Dieses ist am Bildschirm durch Begutachtung jeder Zeichenposition, die durch die "DIRECT CURSOR ADDRESSING" ESC-Folge festgelegt wurde, zu überprüfen. Der Bildschirm muß folgendes Muster anzeigen:

```
CURSOR ADDRESSING TEST
*************
            *
*********** *
```



2.2.6.17. Test 17: REVERSE LINEFEED / INDEX Inverser Zeilenvorschub

Der Bildschirm wird gelöscht und der Cursor auf die HOME-Position bewegt. Auf der obersten Zeile wird der Testname ausgegeben und danach der Kursor auf die unterste Zeile, Spalte 1, bewegt. Dann wird unter Nutzung von "REVERSE LINEFEED" oder "REVERSE INDEX" ESC-Folgen und Ausgeben eines / eine diagonale Linie von unten links nach oben rechts erzeugt.

Beispiel:

```
REVERSE LINEFEED - INDEX TEST CREATES A DIAGONAL LINE
                                    /
                                   /
                                  /
                                 /
                                /
```

2.2.6.18. Test 18: HORIZONTAL TAB Horizontaltabulator

Es wird die Fähigkeit des Terminals zum Setzen und Löschen von Tabulatoren unter Steuerung eines Hostrechners getestet. Der Test besteht aus 2 Teilen.

Teil 1: (Für mechanische Tabulatoren)
Der Bediener wird gefragt, ob die Option vorhanden ist. Wenn YES, wird über die Seite eine Referenzlinie erzeugt, die durch Tabulatorstopps mit 1, 2, 3, ...15 Spalten Abstand zerlegt wird. Drei Zeilen, bestehend aus TAB, I, TAB, I, ..., werden unter die Referenzzeile gedruckt. Die I müssen in einer Linie unter den V in der Referenzzeile stehen.

Beispiel:

```
.V..V...V....V.....V......V...
 I  I   I    I     I      I...
 I  I   I    I     I      I...
 I  I   I    I     I      I...
```

Teil 2: (Für Terminals, die durch ESC-Folgen gesteuert werden)
Eine Refenzzeile, bestehend aus . und V wird über die Seite gedruckt. Dann

werden Tabulatorstopps korrespondierend mit den V in der Refenzzeile gesetzt. Drei Zeilen, bestehend aus TAB, I, TAB, I, ... werden dann unter die Refenzzeile gedruckt. Die I müssen in einer Linie unter den V stehen. Dieser Test wird für Tabulatorabstände von 2, 4, 8, 16 und 32 durchgeführt.

Beispiel für Tabulatorabstand 4:

```
...V...V...V...V...V...
   I   I   I   I   I
   I   I   I   I   I
   I   I   I   I   I
```

2.2.6.19. Test 19: GRAPHICS CHARACTER SET
Grafikzeichensatz

Der Test versetzt das Terminal in den Grafikmodus und schreibt nach Ausgabe der Überschrift

GRAPHICS CHARACTER SET TEST

10 Zeilen mit Zeichen des Grafikzeichensatzes der Kodes 137 bis 176 oktal. MDer Grafiksatz ist für verschiedene Terminals unterschiedlich.

2.2.6.20. Test 20: SCROLLING
Bildschirmrollen

Der Test prüft die Fähigkeit des Terminals, korrekt in den beiden Modi JUMP und SMOOTH SCROLLING zu arbeiten. Der Test füllt 22 Zeilen mit . als Hintergrund. Dann wird ein Rollbereich von Zeile 4 bis 12 definiert. 36 Zeilen eines Musters werden dann gesendet, die nur innerhalb des Rollbereiches erscheinen dürfen und nach oben rollen, wenn der Rollbereich gefüllt ist. Das Muster wird nochmals gesendet unter Nutzung der "REVERSE INDEX" ESC-Folge anstelle von <LF>. In diesem Falle rollt der Bereich nach unten, wenn der Rollbereich gefüllt ist.

Beispiel:

```
SCROLLING TEST.......................
JUMP MODE............................
.....................................
.....................................
         ABCDEFGHIJKL
         BCDEFGHIJKLM
         CDEFGHIJKLMN
         DEFGHIJKLMNO
         EFGHIJKLMNOP
.....................................
.....................................
.
.
```

Nach dem Test des JUMP-Modus erfolgt der gesamte Testablauf noch einmal für den SMOOTH-Modus.

2.2.7. Auflistung der Tests

Test 1: DATA PATH
Datenweg
Test 2: PRINTABLE CHARACTERS
Druckbare Zeichen
Test 3: NON-PRINTABLE CHARACTERS
Nichtdruckbare Zeichen
Test 4: CARRIAGE RETURN
Wagenrücklauf
Test 5: LINE FEED
Zeilenvorschub
Test 6: SPACE / BACKSPACE
Leerzeichen/Rückleerzeichen
Test 7: BELL
Signalton
Test 8: TAB
Tabulator
Test 09: CURSOR HOME
Kursor Rücksetzen
Test 10: CURSOR UP
Kursor nach oben
Test 11: CURSOR RIGHT
Kursor nach rechts
Test 12: LINE ERASE
Zeile löschen
Test 13: SCREEN ERASE
Bildschirm löschen
Test 14: CURSOR LEFT
Kursor nach links
Test 15: CURSOR DOWN
Kursor nach unten
Test 16: DIRECT CURSOR ADDRESSING
Direkte Kursoradressierung
Test 17: REVERSE·LINEFEED / INDEX
Inverser Zeilenvorschub
Test 18: HORIZONTAL TAB
Horizontaltabulator
Test 19: GRAPHICS CHARACTER SET
Grafikzeichensatz
Test 20: SCROLLING
Bildschirmrollen

3. Druckerdiagnoseprogramme

3.1. PVAAA: Test von Druckern mit Parallelinterface

3.1.1. Zusammenfassung

Das Diagnoseprogramm PVAAA ist ein Programm der Ebene 2R (User-Modus, Dok.-Nr. 1.57.700600.7), es läuft unter SVP und nutzt den Druckerdriver. PVAAA ist für den Test der Drucker bestimmt, die an der Druckeransteuerung der AMF18 (K 8081), d.h. mittels Parallelinterface angeschlossen sind. Es testet nicht speziell die Druckeransteuerung der AMF18.
Das Diagnoseprogramm PVAAA besteht aus 13 Tests, die alle die Beobachtung oder den Eingriff des Operators erfordern. Es ist in der Lage, mehrere Drucker nacheinander zu testen. Alle ausgewählten Tests werden erst für einen Drucker ausgeführt, bevor der Test des nächsten Gerätes beginnt. Die Tests erzeugen Druckmuster, die zur Fehlererkennung visuell begutachtet werden müssen. Einige der Muster sind als WORST_CASE-Tests für spezielle Druckertypen entworfen, laufen aber ebenso auf anderen Druckern, da es der Diagnose nicht möglich ist, den angeschlossenen Druckertyp zu erkennen. Durch die Auswahl der Sektionen QUALIFICATION, FUNCTIONAL, MANUAL oder DEFAULT kann der Operator bestimmte Gruppen von Tests auswählen.

3.1.2. Testbedingungen

3.1.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- Anschlußsteuerung AMF18 mit am Parallelausgang angestecktem Drucker

3.1.2.2. Software

- SVP 1800
- Diagnose-Supervisor

3.1.3. Voraussetzungen

- vollständig funktionierender Rechnerkern

3.1.4. Bedienungshinweise

Zuerst ist der Diagnose-Supervisor zu laden und zu starten. Danach wird das zu testende Gerät angeschlossen (ATTACH) und ausgewählt (SELECT). Für den ATTACH-Vorgang gibt es zwei Möglichkeiten: den PROMT-Modus (Eingabeanforderungsmodus) und den EXPLICIT-Modus (Explizitmodus). Im PROMT-Modus wird jede Eingabe einzeln angefordert, im EXPLICIT-Modus wird die gesamte ATTACH-Kommandofolge in drei Zeilen eingegeben. Beispiele für typische ATTACH-Folgen siehe Punkte 3.1.4.1. und 3.1.4.2. Dabei ist die CSR-Adresse 760340 nur ein Beispiel. Die wirkliche CSR-Adresse wird durch Schalter auf der AMF18 bestimmt. Auch für TR, BR, Vektor und Druckertyp sind die aktuellen Werte einzusetzen. Für den Druckertyp (device type) sind einzusetzen:

- DP23 für VT 23X00
- DP27 für VT 270XX

Meist wird der Drucker der Punkt-Queue zugeordnet sein. Dann ist durch das SCL-Kommando STOP/QUEUE/RESET SYS$PRINT die Queue zu stoppen. Dazu ist das OPERATOR-Privileg nötig. Nach dem Kommando SET DEVICE LCA0/NOSPOOLED kann der Drucker attached werden.

3.1.4.1. Kommandos im Eingabeanforderungsmodus

```
DS> LOAD PVAAA                       ;Laden des Testprogramms
DS> ATTACH                           ;Anschluß des SKRBUS-Adapters
Device type?  DWA40                  ;SKRBUS-Adapter
Device Link?  HUB                    ;an ISB
Device Name?  DW0                    ;allg. Gerätename des SKRBUS-Adapters
TR? 3
BR? 4                                ;BUS_REQUEST-Ebene
DS> ATTACH                           ;Anschluß des Controllers
Device type? AMF18P                  ;LP- und DR-Einheit der AMF18
Device Link? DW0                     ;an SKRBUS-Adapter
Device Name? LCA                     ;allg. Gerätename der AMF18P
CSR? 760340                          ;CSR-Adresse (Beispiel)
Vektor? 300                          ;Vektoradresse (Beispiel)
BR? 5                                ;BR-Ebene (Beispiel)
Port Device? LP                      ;Gerät am Port: Paralleldrucker
DS> ATTACH                           ;Anschluß des Druckers
Device type? DP23                    ;Banddrucker VT23600 (Beispiel)
Device Link? LCA                     ;an AMF18
Device Name? LCA0                    ;allg. Gerätename Druckeranschluß
DS> SELECT LCA0                      ;auswählen des Gerätes
```

3.1.4.2. Kommandos in Explizitmodus

Dies ist eine Kurzfassung des Anschließens:

```
DS> LOAD PVAAA
DS> ATT DWA40 HUB DW0 3 4
DS> ATT AMF18P DW0 LCA 760340 300 5 LP
DS> ATT DP23 LCA LCA0
```

3.1.4.3. Starten des Diagnoseprogramms

Nach dem Anschließen (ATTACH) und Auswählen (SELECT) ist die Diagnose zu starten, dafür gibt es zwei Möglichkeiten:

- DS> START/Schalter
- DS> RUN PVAAA/Schalter

Die möglichen Schalter sind der Beschreibung des Diagnose-Supervisors zu entnehmen.

3.1.4.4. Sektionen

Die Tests des Programms PVAAA laufen in vier Sektionen:

DEFAULT:	Tests	2, 3, 5, 6, 7, 8, 9, 11, 12, 15
FUNCTIONAL:	Tests	2, 3, 5, 6, 7, 8, 9, 11, 12, 15
MANUAL:	Tests	29, 31, 32
QUALIFICATION:	Test	2

3.1.4.5. QUICK VERIFY MODE

Im QUICK VERIFY MODE laufen die Tests 2, 3, 8, 9 und 32 entsprechend der Einstellung der Sektion. Der Test 10 wird in den Sektionen DEFAULT und FUNCTIONAL angezeigt, aber nicht abgearbeitet.

3.1.5. Funktionsbeschreibung

3.1.5.1. Übersicht

Das Programm testet den Drucker durch Ausgabe von Testmustern. Die entsprechenden Ausdrucke muß der Operator visuell begutachten.

3.1.5.2. Programmgröße

40960 byte (dezimal)

3.1.5.3. Programmlaufzeit

Die Programmlaufzeit ist abhängig von der Druckgeschwindigkeit des angeschlossenen Druckers. Die Abarbeitung aller DEFAULT-Tests ergibt 13 Seiten mit 69 Zeilen pro Seite. Bei einer Druckgeschwindigkeit von 600 Zeilen/s wäre die Programmlaufzeit ca. 1,5 Minuten.

3.1.6. Testbeschreibungen

3.1.6.1. Test 1: reserviert

3.1.6.2. Test 2: DATA PATH TEST Datenweg

Dieser Test prüft die Datenleitungen über das Interface bis hin zum Drucker. Es werden folgende vier Zeilen ausgegeben, dieses Muster wird noch zweimal wiederholt.

```
*U*U*U*U*U*U*U*U*U
U*U*U*U*U*U*U*U*U*
@?@?@?@?@?@?@?@?@?
?@?@?@?@?@?@?@?@?@
```

Durch diese Kodierungen werden erst zwei Zeilen lang bitweise abwechselnd und von Zeichen zu Zeichen alternierend 1 und 0 auf den Leitungen erzeugt (* = 0101010 U = 1010101), danach zwei Zeilen lang alternierend ein Zeichen alle Bits 1, das nächste alle Bits 0 (@ = 1000000 ? = 0111111).

Abarbeitung im QUICK VERIFY MODE.

Test läuft in den Sektionen QUALIFICATION, FUNCTIONAL und DEFAULT

3.1.6.3. Test 3: PRINTABLE CHARACTER TEST Druckbare Zeichen

Dieser Test prüft alle 95 druckbaren Kodes (oktal 40...176). Er ist ein Zeichentest für Trommel bzw. Band und Zeichengenerator sowie ein Dauertest. Jeder druckbare Kode wird eine volle Zeile lang ausgegeben. Bei Druckern, für die LP$MLOWER nicht gesetzt ist, werden vom Driver Kleinbuchstaben zu Großbuchstaben konvertiert. Wenn die Ansteuerung auf 96 Zeichen eingestellt ist, der Drucker aber nur einen Zeichensatz von 64 Druckzeichen hat, so werden 32 Leerzeilen ausgegeben.

Abarbeitung im QUICK VERIFY MODE.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT

3.1.6.4. Test 4: reserviert

3.1.6.5. Test 5: OVERPRINT TEST Überdrucken

Dieser Test prüft die CARRIAGE_RETURN-Steuerung und damit das Überdrucken und die Wiederholbarkeit der Druckcharakteristik. Das folgende Muster wird ausgegeben, wobei jede Zeile noch einmal überdruckt wird. Das Muster wird viermal wiederholt.

```
M M M M M M M M M
 @ @ @ @ @ @ @ @ @
E E E E E E E E E
```

Test läuft in den Sektionen FUNCTIONAL und DEFAULT

3.1.6.6. Test 6: SPECIAL DRUM PATTERN Spezial-Trommelmuster

Das Muster besteht aus 8 identischen Gruppen mit je 17 Zeichen Breite. Jedes der 17 Zeichen jeder Reihe einer Gruppe hat einen ASCII-Code um eins größer als das vorhergehende Zeichen. Jede neue Reihe jeder Gruppe beginnt mit einem Zeichen, dessen ASCII-Code um eins größer ist, als der des Startzeichens der darüberliegenden Reihe.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT.

3.1.6.7. Test 7: DP27 DRUM PATTERN DP27 Trommelmuster

Das Muster besteht aus 132 Druckzeichen pro Zeile. Jedes Zeichen hat einen ASCII-Kode um drei (3) größer als der des vorhergehenden Zeichens. Jede nachfolgende Zeile beginnt mit einem Zeichen, dessen ASCII-Kode um eins (1) größer ist als der des Startzeichens der vorhergehenden Zeile. Obwohl dieser Test als WORST CASE-Test für Trommeldrucker der Reihe VT27000 entworfen wurde, läuft er auf allen Druckern.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT

3.1.6.8. Test 8: NON-PRINTABLE CHARACTER TEST Nichtdruckbare Zeichen (NPC)

Dieser Test prüft die Kodes, die keine Druckzeichen darstellen und auch keine Steuerfunktion haben. Ein Ausgabepuffer, der aus 7 byte der Mnemonik und N byte mit dem Kode des NPC besteht, wird online gesendet. Dabei gilt N = Druckbreite -7 . Der Operator muß beobachten, daß zwar die Mnemoniks gedruckt, aber ansonsten keine Funktionen ausgelöst werden.

Beispiel:

```
NUL 000
SOH 001
STX 002
```

Abarbeitung im QUICK VERIFY MODE.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT.

3.1.6.9. Test 9: ROTATING CHARACTER PATTERN Rotierende Zeichenmuster

Dieser Test erzeugt ein vollständiges Abbild der Trommel bzw. des Bandes. Er dient als Dauertest und zur Begutachtung der Schallemission. Es wird ein Puffer generiert, der alle 95 druckbaren Zeichenkodes in aufsteigender Reihenfolge enthält (oktal 40...176). Aus diesem Puffer wird eine Zeile ausgegeben, deren Länge der eingestellten Druckbreite entspricht. Der vollständige Puffer wird dann rotierend ein Zeichen nach links verschoben, so daß das vorher erste Zeichen jetzt ans Ende der Zeile kommt. Mit diesem Puffer wird die nächste Zeile gedruckt. Das ganze wiederholt sich 95 mal (jedes Druckzeichen auf jeder Position).

Abarbeitung im QUICK VERIFY MODUS.

Läuft in den Sektionen FUNCTIONAL und DEFAULT.

3.1.6.10. Test 10: reserviert

3.1.6.11. Test 11: LINEFEED TEST Zeilenschaltung

Dieser Test gibt eine Bezugszeile aus, danach eine Folge von Schrägstrichen (SLASH) und Zeilenschaltbefehlen (LINEFEED):
SLASH,LF,SLASH,LF,SLASH ...
Wenn der Drucker ein CARRIAGE RETURN (Wagenrücklauf) benötigt, muß eine diagonale Linie erscheinen. Wenn bei LINEFEED automatisch ein CARRIAGE RETURN eingefügt wird (automatische Rückkehr zu Spalte 1 bei LINEFEED), so müssen die Schrägstriche am linken Rand untereinander stehen, außer in der Bezugszeile, die einen überdruckten Schrägstrich in der Startspalte hat.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT.

3.1.6.12. Test 12: PRINT CONTROL TEST Drucksteuerung

Dieser Test prüft die Druckersteuerlogik, indem mehr als 132 Zeichen gesendet werden ehe das Druckkommando kommt (CR,LF,FF). Der Drucker muß die ersten 132 Zeichen (Druckpufferbreite) korrekt drucken und alle Zeichen nach dem 132sten verwerfen. Es müssen 10 volle Zeilen gedruckt werden, jede Zeile mit einer der Ziffern 0...9.

Test läuft in den Sektionen FUNCTIONAL und DEFAULT.

3.1.6.13. Test 13: reserviert

3.1.6.14. Test 14: reserviert

3.1.6.15. Test 15: SPURIOUS HAMMER FIRINGS
Falsche Hammerschläge

Dieser Test dient der Erkennung falscher Hammerschläge und defekter Hammertreiber. Er erzeugt rechte und linke Keilmuster. Jeder Druck außerhalb des Keils wird durch einen falschen Hammerschlag oder ein Hammerprellen verursacht.

3.1.6.16. Test 16: reserviert

3.1.6.17. Test 17: reserviert

3.1.6.18. Test 18: reserviert

3.1.6.19. Test 19: reserviert

3.1.6.20. Test 20: reserviert

3.1.6.21. Test 21: reserviert

3.1.6.22. Test 22: reserviert

3.1.6.23. Test 23: reserviert

3.1.6.24. Test 24: reserviert

3.1.6.25. Test 25: reserviert

3.1.6.26. Test 26: reserviert

3.1.6.27. Test 27: reserviert

3.1.6.28. Test 28: reserviert

3.1.6.29. Test 29: HAMMER ALIGNMENT TEST
Hammerjustierung

Dieser Test dient der manuellen Einstellung der Hammerflucht und der Anschlagstärke. Er druckt eine ganze Seite mit dem Buchstaben E.

3.1.6.30. Test 30: reserviert

3.1.6.31. Test 31: OPERATOR SELECTED PATTERN
Selbstgewählte Muster

Es können bis zu 132 Zeichen eingegeben werden und diese werden exakt und kontinuierlich zum Drucker geechot, wie sie eingegeben wurden, bis vom Bedienterminal der Wechsel des Musters oder Testende angewiesen wird. Wenn der Operator CTRL/C drückt, kommt er zur Eingabeanforderung DS> zurück. An dieser Stelle kann der Test wieder gestartet oder das Muster geändert werden.

Beispiel:

```
CTRL/C
DS> START/SECTION: MANUAL/TEST:18:18
```

Test läuft in der Sektion MANUAL.

3.1.6.32. Test 32: ERROR BIT TEST
Fehleranzeige

Die Ebene-2-Diagnose kann das ERROR-Bit nicht direkt testen. Aber sie kann das QIO-Kommando nutzen, und wenn das Kommando innerhalb 10 Sekunden nicht abgeschlossen wird, kann angenommen werden, daß der Driver eine Fehlerbedingung erkannt hat. Es wird eine Mitteilung auf dem Bedienterminal ausgegeben die den Operator auffordert, eine Fehlerbedingung zu provozieren (Netz aus, kein Papier, Trommeltor offen, offline usw.) und danach RETURN zu drücken. Die Diagnose wird dann versuchen, die Ausschrift "ERROR NOT DETECTED" auf dem Drucker auszugeben, was aber nicht gelingen dürfte. Daraufhin wird auf dem Bedienterminal "CORRECT THE ERROR CONDITION" ausgegeben.

Abarbeitung im QUICK VERIFY MODE.
Test läuft in der Sektion MANUAL.

3.1.6.33. Test 33: reserviert

3.1.7. Auflistung der Tests

Test 1: reserviert
Test 2: DATA PATH TEST
Datenweg
Test 3: PRINTABLE CHARACTER TEST
Druckbare Zeichen
Test 4: reserviert
Test 5: OVERPRINT TEST
Überdrucken
Test 6: SPECIAL DRUM PATTERN
Spezial-Trommelmuster
Test 7: DP27 DRUM PATTERN
DP27 Trommelmuster
Test 8: NON-PRINTABLE CHARACTER TEST
Nichtdruckbare Zeichen (NPC)
Test 9: ROTATING CHARACTER TEST
Rotierende Zeichenmuster
Test 10: reserviert
Test 11: LINEFEED TEST
Zeilenschaltung
Test 12: PRINT CONTROL TEST
Drucksteuerung
Test 13: reserviert
Test 14: reserviert
Test 15: SPURIOUS HAMMER FIRING
Falsche Hammerschläge
Test 16: reserviert
Test 17: reserviert
Test 18: reserviert
Test 19: reserviert
Test 20: reserviert
Test 21: reserviert
Test 22: reserviert
Test 23: reserviert
Test 24: reserviert
Test 25: reserviert
Test 26: reserviert
Test 27: reserviert
Test 28: reserviert
Test 29: HAMMER ALIGNMENT TEST
Hammmerjustierung
Test 30: reserviert
Test 31: OPERATOR SELECTED PATTERN
Selbstgewählte Muster
Test 32: ERROR BIT TEST
Fehleranzeige
Test 33: reserviert

4. Diagnoseprogramme für Graphische Peripherie

4.1. PVTYA: Test des Plotters K 6411

4.1.1. Zusammenfassung

PVTYA ist ein Diagnoseprogramm der Ebene 2R (User-Modus, Dok.-Nr. 1.57.700612.), das unter SVP 1800 läuft und den Terminal-Driver benötigt.
Das Programm führt den funktionellen Test des Plotters K 6411 durch. Es werden alle in der Befehlsbeschreibung des Plotters angeführten Befehle vom Rechner aus gesendet und eventuelle Antworten ausgewertet. Die Tests sind insofern eingeschränkt, daß wegen der weiten Variabilität der Befehlsparameter und der riesigen Zahl der möglichen Befehlskombinationen nur ausgewählte Befehlsfolgen mit ausgewählten Parametern getestet werden können. Das Diagnoseprogramm dient sowohl der Inbetriebnahme, als auch der Wartung und Fehlersuche. Es ist auch zur Demonstration der Plotterbefehle geeignet.
Vom Rechner auswertbare Fehler werden am Bedienterminal angezeigt.
Der Bediener kann die Sektionen DEFAULT, FUNCTIONAL, QUALIFICATION oder MANUAL auswählen.

4.1.2. Testbedingungen

4.1.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- SKRBUS-Adapter DWA40
- AMF18 mit IFSS-Anschluß

4.1.2.2. Software

- SVP 1800
- Diagnose-Supervisor

4.1.3. Voraussetzungen

- Entsprechend Systemstandard eingestellter Plotter: 9600 Baud, Senden/Empfangen passiv, Parität ungerade, Schirm und Masse getrennt
- Entsprechend Systemstandard eingestellte Terminalleitung zum Plotter
- fehlerfrei abgelaufener Interntest des Plotters
- Bestückung des Stift-Magazins mit 1...7 Stiften entsprechend Betriebsvorschrift. Mindestens einer der Plätze 2...8 muß frei bleiben. Um den Plot exakt beurteilen zu können, sind Stifte mit guter Spitze (am besten Kugelschreiber) einzusetzen.

- Auflegen von Papier:

  dem Default-Skalierungspunkt P1 (0,0).
- Kenntnis der Betriebsvorschrift sowie der Befehlsbeschreibung des Plotters.

4.1.4. Bedienungshinweise

Zuerst ist der Diagnose-Supervisor zu laden und zu starten. Danach wird das Diagnoseprogramm geladen sowie das zu testende Gerät angeschlossen (ATTACH) und selektiert (SELECT):

```
DS> LOAD PVTYA                                        ;Laden PVTYA
DS> ATT DWA40 HUB DW0 3 4  ;Anschluß
                                                      ;SKRBUS-Adapter
DS> ATT AMF18A DW0 TXA 760340 310 5 377 9600 INTERNAL YES ;AMF18A an SKRBUS-
                                                      ;Adapter
DS> ATT GP11 TXA TXA6                                 ;Anschluß Plotter
                                                      ;an AMF18
DS> SEL TXA6                                          ;Auswahl des Plotters
                                                      ;als TXA6
DS> ST                                                ;Start PVTYA
```

Dabei sind die unterstrichenen Angaben nur Beispiele, es sind die Werte entsprechend Anlagenkonfiguration einzustellen.
Nach dem Starten des Programms sollte der Plotter beobachtet werden, um eventuelle Störungen und Fehler, die keine ERROR-Meldungen auslösen (z.B. Fehler beim Fassen oder Ablegen des Stiftes) zu erkennen.

Achtung: Da der benutzte Terminal-Driver keine Zeitüberwachung der Ausgabe hat, bleibt das Programm stehen, wenn ein Hardwarefehler des Plotters (Harderror, Off-line, Plotter aus) während einer Übertragungspause des Drivers auftritt, in der nach DC3 auf DC1 gewartet wird (Übertragung von mehr als 750 Zeichen). Der Bediener muß entweder die Fehlerbedingung beseitigen oder das Programm mittels CRTL/Y abbrechen.
Durch die interne Hard- und Softwarestruktur des Plotters besteht keine eindeutige zeitliche Zuordnung zwischen dem gerade ausgeführten Plot und der Erkennung eines Fehlers auf dem Plotterdisplay oder einer Ausschrift auf dem Bedienerterminal. Aus gleichem Grund kann ein Plot nicht unmittelbar durch CRTL/C abgebrochen werden. Die Daten eines Tests werden an den Plotter gesendet (bis zu 750 Zeichen bei einer Übertragung). Erst wenn alle diese Zeichen vom Eingabepuffer des Plotters zur Ausführung in dessen Abarbeitungspuffer übernommen wurden, ist das Programm wieder unterbrechbar. Dann wird das angemeldete CANCEL ausgeführt.
Soll die Reaktion des Plotters auf einen einzelnen Befehl oder eine spezielle Befehlsfolge geprüft werden, so können diese unabhängig vom Diagnosesystem mit dem SVP-Kommando COPY vom Bedienterminal zum Plotter gesendet werden:

```
z.B.:  COPY TXA1: TXA2:             ;Senden vom TXA1 zum TXA6 (Plotter)
       IN;                          ;Initialisieren
       SP1 PA1000,1000 CI400 SP0;   ;Stift1 holen und auf Position 1000,1000
                                    ;einen Kreis mit 10 mm Radius plotten
```

4.1.5. Funktionsbeschreibung

4.1.5.1. Übersicht

Das Programm führt den funktionellen Test des Plotters K 6411 durch. Es werden alle in der Befehlsbeschreibung des Plotters angeführten Befehle vom Rechner aus gesendet und eventuelle Antworten ausgewertet. Die Tests sind insofern eingeschränkt, daß wegen der weiten Variabilität der Befehlsparameter und der riesigen Zahl der möglichen Befehlskombinationen nur ausgewählte Befehlsfolgen mit ausgewählten Parametern getestet werden können. Das Diagnoseprogramm dient sowohl der Inbetriebnahme, als auch der Wartung und Fehlersuche. Vom Rechner auswertbare Fehler werden am Bedienterminal angezeigt.

4.1.5.2. Programmgröße

Das Programm belegt ca. 58 KByte Speicherplatz.

4.1.5.3. Programmlaufzeit

SECTION DEFAULT:	max. 31 min
SECTION FUNCTIONAL:	max. 8 min
SECTION QUALIFICATION:	max. 6 min
SECTION MANUAL:	max. 13 min

4.1.5.4. Abarbeitungsbedingungen

entfällt;

4.1.5.5. Flagverwendung

entfällt;

4.1.5.6. Allgemeine Programmanforderungen

entfällt;

4.1.5.7. Fehlermitteilung

Es werden alle vom Programm erkannten Fehler gemeldet. Werden fehlerhafte, vom Plotter gesendete Daten empfangen, so werden diese im Hexadezimalcode auf dem Bedienerterminal ausgegeben. Auswertbare Fehlercodes werden kommentiert.
Vier Gruppen von Fehlermeldungen sind zu unterscheiden:

SYSTEM FATAL ERROR: Ein nicht vom Plotter ausgelöster Fehler
DEVICE FATAL ERROR: Ein Plotterfehler, der den sofortigen Programmabbruch erfordert

HARD ERROR: Ein Harderror des Plotters, der Test wird abgebrochen
SOFT ERROR: Ein Syntax- oder Übertragungsfehler, der Test wird fortgesetzt

Innerhalb der Fehlergruppen DEVICE FATAL ERROR, HARD ERROR und SOFT ERROR können folgende Fehlernummern auftreten:

ERROR 1: Zu Beginn des Tests wurde versucht, den Befehl OE auszugeben und auf die Antwort des Plotters gewartet. Der entsprechende QIO kehrt mit Fehlerstatus zurück. STATUS und COUNT des QIO werden ausgeschrieben.

ERROR 2: Als erster Befehl des Tests wurde OE gesendet und auf die Antwort des Plotters gewartet. Die Wartezeit auf die Antwort (3s) wurde überschritten.

ERROR 3: Als erster Befehl des Tests wurde OE gesendet. In der Antwort, dem Fehlerbyte, war das Bit "HARDWAREFEHLER" gesetzt.

ERROR 4: Als erster Befehl des Tests wurde OE gesendet. In der Antwort, dem Fehlerbyte, war eines der Softfehler-Bits (Bits 1...6) gesetzt.

ERROR 5: Nach der Ausgabe von Befehlen an den Plotter, die keine Antwort auslösen, wurde versucht, den Befehl OE auszugeben und auf die Antwort des Plotters gewartet. Der entsprechende QIO kehrt mit Fehlerstatus zurück. STATUS und COUNT des QIO werden ausgeschrieben.

ERROR 6: Nach der Ausgabe von Befehlen an den Plotter, die keine Antwort auslösen, wurde der Befehl OE gesendet und auf die Antwort des Plotters gewartet.Die Wartezeit auf die Antwort (3s) wurde überschritten.

ERROR 7: Nach der Ausgabe von Befehlen an den Plotter, die keine Antwort auslösen, wurde der Befehl OE gesendet. In der Antwort, dem Fehlerbyte, war das Bit "HARDWAREFEHLER" gesetzt.

ERROR 8: Nach der Ausgabe von Befehlen an den Plotter, die keine Antwort auslösen, wurde der Befehl OE gesendet. In der Antwort, dem Fehlerbyte, war eines der Softfehler-Bits (Bits 1...6) gesetzt.

ERROR 9: Es wurde versucht, Befehle an den Plotter zu senden, die keine Antwort auslösen. Der entsprechende QIO kehrt mit Fehlerstatus zurück. Der QIO-STATUS wird ausgeschrieben.

ERROR 10: In der Antwort auf einen Ausgabebefehl (Ox) wurden mehr als 100 Gerätestatusmeldungen DC4 erkannt.

ERROR 11: In der Antwort auf einen Ausgabebefehl (Ox) wurde als erstes Nutzzeichen (nach eventuellen DC4) das Abschlußzeichen <CR> erkannt.

ERROR 12: In der Antwort auf einen Ausgabebefehl (Ox) wurde statt der erwarteten Ziffer im ASCII-Code ein anderes Zeichen erkannt.

ERROR 13: Es wurde versucht, einen Ausgabebefehl (Ox) auszugeben und auf die Antwort des Plotters gewartet. Der entsprechende QIO kehrt mit Fehlerstatus zurück. STATUS und COUNT des QIO werden ausgeschrieben.

ERROR 14: Es wurde versucht, einen Ausgabebefehl (Ox) auszugeben und auf die Antwort des Plotters gewartet. Die Wartezeit auf die Antwort wurde überschritten.

ERROR 15: Beim Vergleich von empfangenen mit erwarteten Daten trat ein Differenz auf. Die erwarteten und die empfangenen Daten werden auf dem Bildschirm angezeigt.

ERROR 16: Im Test 6 wurde mittels Befehl OT die Belegung der Stiftbox abgefragt. Dabei wurde die Antwort nicht ordnungsgemäß mit "0," eingeleitet.

4.1.5.8. Programmitteilungen

entfällt;

4.1.5.9. Leistungsparameter des Testobjekts

entfällt;

4.1.5.10. Aktionen bei unerwarteten Fehlern

entfällt;

4.1.5.11. Testfolge bei mehreren Einheiten

entfällt;

4.1.5.12. Einstellungen für Ablauf unter AFT

entfällt;

4.1.5.13. Testsektionen

DEFAULT:	Tests 1...39 Test aller Plotterfunktionen außer Digitalisieren
FUNCTIONAL:	Tests 1...13 Test der wichtigsten Befehle
QUALIFICATION:	Tests 27...31 Tests mit hoher mechanischer Belastung
MANUAL:	Tests 40...41 Test Digitalisieren, Test Senden ohne Ox-Befehle

4.1.5.14. Verschiedenes

entfällt;

4.1.6. Testbeschreibungen

Alle Tests (außer Test 41) beginnen mit dem Senden des Abfragebefehls OE und des Befehls DF. Danach wird in allen Tests mit Plot (Tests 2...31, 40, 41) der Stift 1 geholt, ein rechteckiger Rahmen um die vom jeweiligen Test auf dem Papier belegte Fläche gezogen und in die rechte untere Ecke des Testrahmens die Nummer des Tests geplottet. Dabei müssen die Rahmenlinien aneinandergrenzender Tests genau aufeinander liegen.
Am Ende jedes Tests mit Plot wird der Stift wieder in der Box abgelegt. Dann wird der Befehl DF gesendet. Zum Abschluß des Tests (außer Test 41) wird wiederum mit dem Befehl OE der Fehlerstatus ausgewertet.
Ohne Skalierung arbeiten die Tests 1, 2, 5, 25, 26, 32...34, 36, 37 und 40.

4.1.6.1. Test 1: INITIALIZE
Initialisieren

Es wird nur der Befehl IN ausgegeben.

4.1.6.2. Test 2: VECTORS & CIRCLE
Vektoren und Kreis

Es werden zwei Dreiecke geplottet, die ein diagonal geteiltes Quadrat bilden. Das Dreieck oben links wird mittels PA, das unten rechts mit PR geplottet. Daneben werden mittels CI drei ineinanderliegende Kreise geplottet.

4.1.6.3. Test 3: ARC ABSOLUTE & CHORD TOLERANCE
Absoluter Kreisbogen und Toleranzart für Kreisbögen

Mit dem Befehl AA wird oben beginnend ein Kreisbogen 270 Grad entgegen dem Uhrzeigersinn geplottet. Nach CT1 werden 270 Grad eines Sechsechs und nach CT werden mit AA und Winkeltoleranz = 120 Grad 2/3 eines Dreiecks in gleichem Sinn geplottet.

4.1.6.4. Test 4: ARC RELATIVE & CHORD TOLERANCE
Relativer Kreisbogen und Toleranzart für Kreisbögen

Mit dem Befehl AR wird oben beginnend ein Kreisbogen 270 Grad im Uhrzeigersinn geplottet. Nach CT1 werden 270 Grad eines Sechsechs und nach CT werden mit AR und Winkeltoleranz = 120 Grad 2/3 eines Dreiecks in gleichem Sinn geplottet.

4.1.6.5. Test 5: CHARACTER SET 0
Zeichensatz 0

Der Zeichensatz 0 wird mit Standardeinstellung geplottet.

4.1.6.6. Test 6: SELECT PEN
Auswahl des Zeichenwerkzeugs

Es wird ein Feld mit zweimal vier Boxplätzen geplottet, danach mit OT die Magazinbelegung abgefragt. Jeder Stift wird geholt und seine Boxnummer in das entsprechende Fach geplottet.

4.1.6.7. Test 7: IDENTFICATION
Ausgabe der Plotteridentifikation

Nach der Ausgabe des Schriftfeldes: robotron
RVS K 1840
Diagnose-
programm PVTYA
wird mit OI die Plotteridentifikation abgefragt und geplottet.

4.1.6.8. Test 8: DIRECTION ABSOLUTE
Absolute Plotrichtung

Ausgehend vom Mittelpunkt des Testrahmens werden die Befehle "DI 1, 0", "DI 1, 1" usw. bis "DR 1,-1" entsprechend ihren Richtungen geplottet.

4.1.6.9. Test 9: WINDOW & PLOT
Plotfenster und Vektoren

Ein durch kleine Winkel in den Ecken gekennzeichnetes Plotfenster zeigt einen Ausschnitt aus einem wesentlich größeren schraffierten Dreieck.

4.1.6.10. Test 10: SCALE
Achsen

An die Koordinatenachsen x und y werden 6 mm lange Skalenstriche angetragen, je zwei in Plus-Richtung, mittig und in Minus Richtung. Durch 1 mm lange Skalenstriche werden die Koordinatenachsen zum Quadrat ergänzt.

4.1.6.11. Test 11: LINE TYPES
Linientypen

Alle Linientypen werden mit dem entsprechenden Parameter des LT-Befehls geplottet.

4.1.6.12. Test 12: SYMBOL MODE
Symbolmodus

Ein großes W wird mit Vollinie geplottet. An den Eckpunkten wird durch SM ein * erzeugt. Ein großes M wird mit Strichpunktlinie geplottet. An den Eckpunkten

wird durch SM ein O geplottet. Darunter erscheinen an den Enden von Vektoren, die mit abgehobenem Stift ausgegeben werden, neun # nebeneinander.

4.1.6.13. Test 13: LABEL TERMINATOR Terminator für LB-Befehle

Der alle Buchstaben enthaltende Testsatz "The quick brown fox jumps over the lazy dog" wird viermal untereinander ausgegeben. Dabei werden nacheinander mit DT die Terminatoren g, z, v und w gesetzt und die Sätze nach dem jeweiligen Terminator abgebrochen.

4.1.6.14. Test 14: LABEL ORIGIN Positionierung von Zeichenketten

Alle LO-Befehle werden geplottet, ein kleiner Kreis markiert die aktuelle Stiftposition vor dem LB-Befehl.

4.1.6.15. Test 15: DIRECTION RELATIVE Relative Plotrichtung

Ausgehend vom Mittelpunkt des Testrahmens werden die Befehle "DR 1, 0", "DR 1, 1" usw. bis "DR 1,-1" entsprechend ihren Richtungen geplottet.

4.1.6.16. Test 16: WINDOW & LABEL Plotfenster und Plot von Zeichen

Ein durch kleine Winkel in den Ecken gekennzeichnetes Plotfenster zeigt den LB-String "WINDOW" in verschieden Größen: oben im Standardformat voll, in der Mitte so vergrößert, daß die Seiten geclippt wurden und unten noch größer an beiden Seiten und unten geclippt.

4.1.6.17. Test 17: SLANT Kursivschrift

Das Wort "SLANT" wird mit unterschiedlicher Buchstabenneigung geplottet: oben von stark linksgeneigt bis senkrecht, darunter von senkrecht bis stark rechtsgeneigt.

4.1.6.18. Test 18: USER DEFINED CHARACTER Definition Nutzerspezifischer Zeichen

Links erscheint das Summenzeichen in einer fixen Zeichenbox definiert, rechts daneben in einer variablen Zeichenbox. Beide Sigma müssen identisch sein.

4.1.6.19. Test 19: INSERT SCALING POINTS
Setzen der Skalierungspunkte

Innerhalb des Testrahmens werden vier Bilder dadurch erzeugt, daß das gleiche Datenfeld (ein Quadrat mit einem Geradeaus/Linksabbiege-Pfeil und der diagonalen Schrift "K 6411") mit vier unterschiedlichen Skalierungspunkt-Paaren ausgegeben wird. Die Skalierungspunkte werden mitgeplottet.

4.1.6.20. Test 20: SCALING
Skalierung des Plotbereichs

Innerhalb des Testrahmens werden vier Bilder dadurch erzeugt, daß nach dem Plotten des durch die jeweiligen Skalierungspunkte festgelegten Quadrats das gleiche Datenfeld (ein Quadrat mit einem eingeschriebenen CI-Kreis und einem gestrichelten Kreis, dessen obere Hälfte mit AA und die untere mit AR erzeugt wurde) mit unterschiedlicher Skalierung ausgegeben wird:

(Bezogen auf oben links):

oben links: x/y = 1/1
unten links: x/y = 2/1
oben rechts: x/y = 1/2
unten rechts: x/y = 2/2

4.1.6.21. Test 21: SIZE ABSOLUTE CHARACTER
Absolute Zeichengröße

Es werden drei gestrichelte Rechtecke geplottet. Die ausgegebenen Strings müssen die Rechtecke genau ausfüllen.

4.1.6.22. Test 22: SIZE RELATIVE CHARACTER
Relative Zeichengröße

Es werden drei gestrichelte Rechtecke geplottet. Die ausgegebenen Strings müssen die Rechtecke genau ausfüllen.

4.1.6.23. Test 23: CHARACTER CORD TOLERANCE
Rundungsgrad für Zeichen

Der Buchstabe P wird zweimal ausgegeben: links nach CC45, rechts nach CC.

4.1.6.24. Test 24: CARRIAGE RETURN PLOT
Leerzeichen und -zeilen

Es werden die Buchstaben A und Z diagonal versetzt in den Testrahmen geplottet. Nach Plotten eines kleinen Kreises am Fußpunkt des A werden mehrere CP-Befehle mit unterschiedlichen Parametern ausgegeben. Der danach geplottete Kreis muß genau am Fußpunkt des Z sitzen.

4.1.6.25. Test 25: CHARACTER SET 20
Zeichensatz 20

Mit Standardeinstellung wird der Zeichensatz 20 (kyrillisch) ausgegeben.

4.1.6.26. Test 26: CHARACTER SET 10
Zeichensatz 10

Mit der Einstellungen "SI 0.7,0.8" wird der Zeichensatz 10 ausgegeben.

4.1.6.27. Test 27: PLOT DIRECTIONS
Plot in alle Richtungen

Es werden Vektoren in alle Richtungen von 0 bis 360 Grad geplottet, Schrittweite ist 5 Grad. Von 0 bis 180 Grad unten in der Mitte beginnend, 180...360 Grad ab oben Mitte.

4.1.6.28. Test 28: HIGH SPEED & HIGH ACCELERATION
Hohe Geschwindigkeit und
hohe Beschleunigung

Nach Einstellen der maximalen Geschwindigkeit und Beschleunigung werden erst vier Kreise von 6 mm Durchmesser geplottet, die dann durch Vektoren verbunden werden. Die Knickpunkte der Vektoren müssen in den Kreismitten liegen.

4.1.6.29. Test 29: CIRCLES & ARCS
Kreise und Kreisbögen

Zur Überprüfung mit dem Zirkel werden Kreise und Kreisbögen um den markierten Mittelpunkt geplottet. Der äußere CI-Kreis mit Radius 55 mm wird mit maximaler Beschleunigung und Geschwindigkeit geplottet. Mit Standardbedingungen folgt der CI-Kreis mit 50 mm Radius. Die vier Kreisbögen haben einen Radius von 45 mm. Die beiden oberen sind mit AA jeweils erst gegen und dann im Uhrzeigersinn, die unteren mit AR erst im dann gegen den Uhrzeigersinn geplottet. Der gestrichelte CI-Kreis hat einen Radius von 40 mm. Die Markierung des Mittelpunkts ist auch Zielpunkt der Routine 40.

4.1.6.30. Test 30: VELOCITIES
Geschwindigkeiten
des Zeichenwerkzeuges

Die Beschleunigung ist auf Maximum eingestellt. Es werden waagerechte Zickzacklinien von 95 mm Länge untereinandergeplottet, deren Spitzen jeweils 2 mm voneinander entfernt sind. Beginnend mit VS1 wird nach jeweils acht Zickzacks der VS-Parameter auf 10, 20, 30, 40, 50 und 60 gestellt. Das Weiterschalten des VS-Parameters kann auf dem Plotterdisplay beobachtet werden.
Rechts daneben wird dann mit VS60 eine Zickzacklinie von 10 mm Länge mit einem Spitzenabstand von 1 mm geplottet.

4.1.6.31. Test 31: FORCES
Auflagekräfte
des Zeichenwerkzeuges

Es werden senkrechte Zickzacklinien geplottet. Der Parameter des Befehls für die Auflagekraft wird mit FS1 beginnend nach jeweils vier Zickzaks um 1 erhöht. Das Weiterschalten des FS-Parameters kann auf dem Plotterdisplay beobachtet werden.

4.1.6.32. Test 32: ERROR PROVOCATION
Fehlerprovokation

Durch Ausgabe fehlerhafter Befehle werden nacheinander die Softfehler "Sprachbefehl nicht erkannt", "falsche Anzahl von Parametern", "falsche Parameter" und "unbekannter Zeichensatz" provoziert und mittels OE-Befehl die Reaktion des Plotters abgefragt. Da die Abfrage die Display-Anzeige löscht, sieht man die Fehleranzeige an Plotter nur jeweils kurz aufblinken.

4.1.6.33. Test 33: INPUT MASK
Maske für Fehleranzeige

Es wird die Fehlermaske IM0 ausgegeben und überprüft, daß bei einer Fehlerprovokation analog Test 32 kein Fehler gemeldet wird.

4.1.6.34. Test 34: OUTPUT ACTUAL PEN STATUS
Ausgabe der physischen
Position des Zeichenwerkzeugs

Ausgabe einer Wagenbewegung bis außerhalb eines Fensters, Abfrage und Vergleich der gemeldeten Wagenposition.

4.1.6.35. Test 35: OUTPUT COMMANDED PEN STATUS
Ausgabe der logischen
Position des Zeichenwerkzeugs

Ausgabe einer Wagenbewegung bis außerhalb eines Fensters, Abfrage und Vergleich der gemeldeten Wagenposition.

4.1.6.36. Test 36: OUTPUT FACTORS
Ausgabe der Plottereinheiten
pro Millimeter

Ausgabe des OF-Befehls und Prüfung der Antwort.

**4.1.6.37. Test 37: OUTPUT HARD-CLIP LIMITS
Ausgabe der Koordinaten
des Plotbereichs**

Ausgabe des OH-Befehls und Prüfung der Antwort.

**4.1.6.38. Test 38: OUTPUT P1 & P2
Ausgabe der Skalierungspunkte**

Ausgabe von Skalierungspunkten, Ausgabe des OP-Befehls und Prüfung der Antwort.

**4.1.6.39. Test 39: OUTPUT STATUS
Ausgabe des Plotterstatus**

Ausgabe des Initialisierungsbefehls und Prüfung der Antwort auf OS.

**4.1.6.40. Test 40: DIGITIZE POINT & OUTPUT
DIGITIZED POINT
Digitalisieren eines Punkes und
Ausgabe dessen Koordinaten**

Der Stift wird auf eine bestimmte Position gefahren (Mittelpunkt von Test 29), dort wird ein winziger Strich von 0.5 mm Länge geplottet. Der Bediener wird am Bedienterminal aufgefordert, das Plotterdisplay zu beobachten. Wenn dort die Anzeige "dP" erscheint ist die ENTER-Taste des Plotters (und nur diese !!) zu drücken. Damit wird gemeldet, daß der digitalisierte Punkt übernommen werden kann. Auf dem Plotter-Display erscheint "dS" und der Test wird abgeschlossen.

**4.1.6.41. Test 41: TESTS 2...13 WITHOUT INPUT_PROMPTS
Tests 2...13 ohne
Plotter-Ausgabebefehle (Ox)**

Sollte aus irgendeinem Grunde die Ausgabe vom Plotter zum Rechner nicht funktionieren, so können mit Test 41 die FUNCTIONAL-Tests, aus denen alle Plotterausgabe-Befehle entfernt wurden, abgearbeitet werden. Damit wird ein reines Senden zum Plotter verwirklicht.

4.1.7. Auflistung der Tests

Test 1: INITIALIZE
Initialisieren
Test 2: VECTORS & CIRCLE
Vektoren und Kreis
Test 3: ARC ABSOLUTE & CHORD TOLERANCE
Absoluter Kreisbogen und
Toleranzart für Kreisbögen
Test 4: ARC RELATIVE & CHORD TOLERANCE
Relativer Kreisbogen und

Toleranzart für Kreisbögen
Test 5: CHARACTER SET 0
Zeichensatz 0
Test 6: SELECT PEN
Auswahl des Zeichenwerkzeugs
Test 7: IDENTIFICATION
Ausgabe der Plotteridentifikation
Test 8: DIRECTION ABSOLUTE
Absolute Plotrichtung
Test 9: WINDOW & PLOT
Plotfenster und Vektoren
Test 10: SCALE
Achsen
Test 11: LINE TYPES
Linientypen
Test 12: SYMBOL MODE
Symbolmodus
Test 13: LABEL TERMINATOR
Terminator für LB-Befehle
Test 14: LABEL ORIGIN
Positionierung von Zeichenketten
Test 15: DIRECTION RELATIVE
Relative Plotrichtung
Test 16: WINDOW & LABEL
Plotfenster und Plot von Zeichen
Test 17: SLANT
Kursivschrift
Test 18: USER DEFINED CHARACTER
Definition Nutzerspezifischer Zeichen
Test 19: INSERT SCALING POINTS
Setzen der Skalierungspunkte
Test 20: SCALING
Skalierung des Plotbereichs
Test 21: SIZE ABSOLUTE CHARACTER
Absolute Zeichengröße
Test 22: SIZE RELATIVE CHARACTER
Relative Zeichengröße
Test 23: CHARACTER CORD TOLERANCE
Rundungsgrad für Zeichen
Test 24: CARRIAGE RETURN PLOT
Leerzeichen und -zeilen
Test 25: CHARACTER SET 20
Zeichensatz 20
Test 26: CHARACTER SET 10
Zeichensatz 10
Test 27: PLOT DIRECTIONS
Plot in alle Richtungen
Test 28: HIGH SPEED & HIGH ACCELERATION
Hohe Geschwindigkeit und
hohe Beschleunigung
Test 29: CIRCLES & ARCS
Kreise und Kreisbögen
Test 30: VELOCITIES
Geschwindigkeiten
des Zeichenwerkzeuges
Test 31: FORCES
Auflagekräfte
des Zeichenwerkzeuges

Test 32: ERROR PROVOCATION
Fehlerprovokation
Test 33: INPUT MASK
Maske für Fehleranzeige
Test 34: OUTPUT ACTUAL PEN STATUS
Ausgabe der physischen
Position des Zeichenwerkzeugs
Test 35: OUTPUT COMMANDED PEN STATUS
Ausgabe der logischen
Position des Zeichenwerkzeugs
Test 36: OUTPUT FACTORS
Ausgabe der Plottereinheiten
pro Millimeter
Test 37: OUTPUT HARD-CLIP LIMITS
Ausgabe der Koordinaten
des Plotbereichs
Test 38: OUTPUT P1 & P2
Ausgabe der Skalierungspunkte
Test 39: OUTPUT STATUS
Ausgabe des Plotterstatus
Test 40: DIGITIZE POINT & OUTPUT
DIGITIZED POINT
Digitalisieren eines Punkes
und Ausgabe dessen Koordinaten
Test 41: TESTS 2...13 WITHOUT INPUT_PROMPTS
Tests 2...13 ohne
Plotter-Ausgabebefehle (Ox)

4.2. PVTZA: Test des DG K 6404.20

4.2.1. Zusammenfassung

Das DG K 6404.20 ist an einen asynchronen Kanal der AMF18 angeschlossen. Das Testprogramm (Dok.-Nr. 1.57.700611.1) beinhaltet den Funktionstest unter Nutzung der DG-internen Tests und den Test als 'WORKSTATION'. Es arbeitet im Supervisor-Modus 2R (user modus) und nutzt den Terminal-Driver des Betriebssystems SVP 1800. Zur besseren Auswertung der einzelnen Tests werden dem Operator alle Aktionen mitgeteilt.

4.2.2. Testbedingungen

4.2.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- SKRBUS-Adapter DWA40
- AMF18 mit IFSS Anschluß

4.2.2.2. Software

- SVP 1800 mit Diagnose-Supervisor
- Datei PVTZA.DAT (internes DG-Programm für GKS)

4.2.3. Voraussetzungen

- funktionsfähiges RVS K 1840
- funktionsfähiger SKRBUS-Adapter

4.2.4. Bedienungshinweise

Zum prinzipiellen Funktionsnachweis sind folgende Bedienhandlungen notwendig:

```
DS> LOAD PVTZA                                  ;LADEN TESTPROGRAMM
DS> ATT DWA40 HUP DWA0 3 4                      ;ANSCHLU/S SKRBUS-ADAPTER
DS> ATT AMF18 DW0 TXA 760400 310 5 377 9600 INTERNAL YES
                                                ;ANSCHLU/S AN AMF18
DS> ATT GD04 TXA TXA2
DS> SEL TXA2
DS> ST                                          ;START VON TEST1 UND TEST2
```

Mit den Tests 1 und 2 wird die Verfügbarkeit des DG K 6404.20 an der AMF18 geprüft. Diese Tests sind auch in den Sektionen FUNCTIONAL und QUALIFICATION enthalten. Die Ergebnisse der Tests (interner DG-Test) werden dem Operator am Bildschirm mitgeteilt. Das Unterdrücken der Mitteilungen erfolgt über SET EVENT FLAG 1.
Für die Arbeit des DG K 6404.20 als 'WORKSTATION' muß dieses mit der Datei PVTZA.DAT (internes DG-Programm) geladen werden. Das Laden erfolgt mit dem Test 3 (Sektion MANUAL) und setzt das Vorhandensein dieser Datei voraus. In der

Sektion sind noch andere Tests im Zustand realisiert.
Bei einem auftretenden Fehler wird das Testprogramm beendet, da es im DG K 6404.20 durch fehlerhafte Abläufe zu "Verklemmungen" kommen kann (intervallmäßiges Hupen). In einem solchen Fall muß das DG K 6404.20 über Netz Aus-Ein wieder in den Grundzustand gebracht und neu geladen werden.

4.2.5. Funktionsbeschreibung

4.2.5.1. Übersicht

Das Testprogramm beinhaltet vier Sektionen:

QUALIFICATION	)	Test 1 und 2
FUNCTIONAL	)	Auswertung der internen Tests
DEFAULT	)	
MANUAL	)	Laden mit PVTZA.DAT sowie Tests einfacher GKS-Funktionen (kein Digitalisieren)

4.2.5.2. Programmgröße

PVTZA benötigt ca. 20 Kbyte Speicherplatz.

4.2.5.3. Programmlaufzeit

Im schnellen Durchlauf (QUICK) werden nur die Tests 1 und 2 abgearbeitet. Diese brauchen ca. 10 sec.

4.2.5.4. Abarbeitungsbedingungen

entfällt

4.2.5.5. Flag-Verwendung

Es werden das QUICK-Flag und das EVENT-FLAG 1 in den Tests 1 und 2 ausgewertet. Bei gesetztem EVENT-FLAG 1 werden die Bedienerausschriften unterdrückt.

4.2.5.6. Allgemeine Programmanforderungen

Für die Einstellung der Terminalleitung gilt:

- NOWRAP
- PATHRU
- NOTYPE_AHAED
- NOECHO
- DEVICE_TYPE:GD04
- NOEIGHTBIT

- PARITY ODD

4.2.5.7. Fehlermitteilungen

Es werden 3 Fehlerquellen unterschieden:

- Fehler, die der Terminal Driver erkennt (QIO SYSTEM ERROR)
- Zeitfehler beim Lesen vom DG K 6404.20
- fehlerhafte Stringfolgen vom DG K 6404.20

4.2.5.7.1. QIO SYSTEM ERROR

Diese Fehler werden im Rückkehrcode des Drivers bereitgestellt. Sie deuten auf einen prinzipiellen Fehler im SVP 1800, d.h. der Nutzer hat keine Testprivilegien oder ein vorheriger QIO wurde nicht abgeschlossen.

4.2.5.7.2. Zeitfehler

Alle vom DG K 6404.20 zu erwartenden Antworten werden zeitlich überwacht (3 sec) Kommt es zu einem TIME OUT erfolgt folgende Meldung:

z.B im Test 1:

```
output 'R' (available unit):
********  DG-TESTPROGRAMM - 1.0  ********
Pass 1, Test 1, Subtest 0, Error 3, 30-SEP-1987 14:32:17.55
DEVICE FATAL ERROR WHILE TESTING TXA4:  TIME OUT ON READ

********  END OF DEVICE FATAL ERROR NUMBER 3  ********

>>> NUMBER OF REC.DATAS: 0
```

Die weiteren Mitteilungen sind programmspezifisch und haben für den Bediener keine Bedeutung. Diese ergeben sich aus dem angewiesenen Programmabruch.

Diese Meldung kann folgende Ursachen haben:

- Die Leitungsparameter (Geschwindigkeit,Bit je Zeichen,Parität) stimmen nicht überein
- Das interne DG Programm ist in einem undefinierten Zustand. In diesem Fall muß das Gerät über Netz Aus-Ein neu gestartet werden
- Falsche Leitung gewählt oder Leitung defekt

4.2.5.7.3. Fehlermitteilungen durch PVTZA

Diese Meldungen ergeben sich aus der Auswertung der Antwortfolgen des DG K 6404.20. Prinzipiell werden alle Tests dem Bediener durch Ausschriften kommentiert. Aus diesen kann der Bediener die Art des Fehlers und den aktuellen

Testzustand erkennen. Fehlerhafte Antwortfolgen werden hexadezimal mitgeteilt und sind selbsterklärend.

4.2.5.8. Programmitteilungen

Dem Bediener werden alle Testschritte durch Ausschriften kommentiert. Vom Bediener werden Gerätekenntnisse vorausgesetzt. Ein Unterdrücken dieser Ausschriften ist nur im Test 1 und 2 durch Setzen von EVENT-FLAG 1 möglich.

4.2.5.9. Leistungsparameter der Testobjekte

entfällt

4.2.5.10. Aktionen bei unerwarteten Fehlern

Das DG K 6404.20 ist über Netz Aus-Ein neu zu starten und neu zu laden.
Das interne Programm geht bei Netzausfall verloren.

4.2.5.11. Testfolge bei mehreren Einheiten

entfällt

4.2.5.12. Einstellungen für Ablauf unter AFT

entfällt

4.2.5.13. Testsektionen

DEFAULT	)	Auswertung der internen
QUALIFICATION	)	Tests
FUNCTIONAL	)	Test 1 und Test 2
MANUAL	)	Test von GKS-Funktionen
	)	Test 3,4,5

4.2.5.14. Verschiedenes

entfällt

4.2.6. Testbeschreibungen

PVTZA umfaßt insgesamt 5 Tests, die auf 4 Sektionen aufgeteilt sind (siehe Pkt.4.5.13). Diese Aufteilung wurde nach der Art des Testes bzw. Testfolgen vorgenommen.

4.2.6.1. Test 1: SINGLE TEST
Einfachtest

4.2.6.2. Test 2: CONTINIOUS TEST
Fortlaufender Test

Diese beiden Tests starten die internen DG-Tests und werten die Antworten aus. Das DG K 6404.20 muß dabei im Zustand 'METAFILE' (Zustand nach Netz-Ein) sein. Die gesendeten Testfolgen sind vorgeschrieben. Zur besseren Übersicht werden dem Bediener die Ergebnisse der Testfolgen mitgeteilt. Im Test 1 werden Testergebnisse akustisch mitgeteilt (hoher Ton für fehlerfrei - tiefer Ton für Fehler). Sollte ein Fehler erkannt werden, wird der fehlerhafte String hexadezimal mitgeteilt und PVTZA abgebrochen.
Der Bediener muß das DG K 6404.20 über Netz Aus-Ein neu starten.

Die Mitteilungen an den Bediener haben folgende Form:

Test 1:

```
               >>> start of single Test <<<

Output 'R' (available unit): answer OK
Output 'T' ( single test  ): high sound for OK else deep sound for ERROR
Output  64 x '3'           : high sound for OK else deep sound for ERROR
Output 'K' (char for test ): 1.K  2.K  3.K  4.K  - OK
Output 'S' ( stop DG      ): none answer
```

Test 2:

```
               >>> start of continius test <<<

Output 'R' (avalaible unit ): answer OK
Output 'D' ( continius test): answer OK
Output  64 x '3'            : answer OK
Output 'K' ( char for test ): 1.K  2.K  3.K  4.K  - OK
```

4.2.6.3. Test 3: LOADING PVTZA.DAT
Laden mit PVTZA.DAT

Dieser Test läuft nur in der Sektion 'Manual'.
Hiermit kann das DG K 6404.20 mit der Datei PVTZA.DAT geladen und als WORKSTATION betrieben werden. Vor dem Laden wird der Zustand abgefragt und bei ungeladen die Ladeprozedur angesprochen andernfalls nur der aktuelle Zustand mitgeteilt.
Nach dem fehlerfreien Laden sind die zwei Testpunkte P1 und P2 zu digitalisieren.
Der Bediener wird dazu über eine Mitteilung aufgefordert und muß diese mit OK quittieren, wenn das Digitalisieren beendet ist. Diese Bedienhandlung ist genau

einzuhalten, da es sonst zu Fehlern im weiteren Testablauf kommt. Bei fehlerhaftem Digitalisieren von P1 und P2 ist über die Taste 'U' der Lupe der Vorgang zu wiederholen. Bei einem fehlerhaften Laden ist das DG K 6404.20 über Netz Aus-Ein neu zu starten und der Ladevorgang zu wiederholen. Das Laden dauert ca. 1 min. Folgende Meldungen werden ausgeschrieben:

>>> loading DG with file PVTZA.DAT <<<

loading state: DG is no loaded

loading DG - please wait 1 min : loading OK

*** digitizing of points T1 and T2 then enter 'OK':

loading state: DG is loaded and 'OPEN'

4.2.6.4. Test 4: ERROR PROVOCATION
Fehlerprovokation

In diesem Test wird eine falsche Anweisung gesendet und der Fehlerstatus abgefragt. Die Antwort des DG K 6404.20 wird hexadezimal mitgeteilt und hat folgende Form:

>>> error provocation <<<

rec.datas (hex): 02 40 41 24 4D 48 24 52 17 00

error spezification: 0 1 141

4.2.6.5. Test 5: LOADING STATE
Ladezustand.

Es werden der aktuelle GKS-Zustand und die Kennung abgefragt. Bei GKS OPEN wird danach in den Zustand GKS CLOSE geschaltet; bei GKS CLOSE wird nach GKS OPEN geschaltet und bei 'ungeladen' wird eine Rücksetzfolge (STX CAN ETX) gesendet, die das DG K 6404.20 wieder in den Zustand 'METAFILE' setzt. Das Rücksetzen aus dem GKS-Zustand nach 'METAFILE' ist gerätetechnisch nicht möglich. Die Mitteilung hat z.B. folgende Form:

>>> request of loading and identification code <<<

loading state: DG is loaded and OPEN

set DG from state "OPEN" to state "CLOSE"

identification code: STX ESC 110 ETB CR

loading state: DG is loaded and "CLOSE"

4.2.7. Auflistung der Tests

Test 1: SINGLE TEST
Einfachtest
Test 2: CONTINIOUS TEST
Fortlaufender Test
Test 3: LOADING PVTZA.DAT
Laden mit PVTZA.DAT
Test 4: ERROR PROVOCATION
Fehlerprovokation
Test 5: LOADING STATE
Ladezustand

5. D i a g n o s e p r o g r a m m e   f ü r   M a g n e t b a n d t e c h n i k

5.1. PVMZA: Datenzuverlässigkeits-diagnoseprogramm

5.1.1. Zusammenfassung

Das Diagnoseprogramm PVMZA (Dok-Nr. 1.57.700613.6) ist für Kundendienstzwecke zum Test von ISOT 1070 C Magnetbandsubsystemen entwickelt worden. Es ist ein Diagnoseprogramm der Ebene 2, welches dem Operator erlaubt, die Bandlaufwerke sowohl mit als auch ohne SVP zu testen.
Das Programm ist fähig bis zu vier Magnetbandlaufwerke CM 5306/5308/5309 über ihren Kontroller ISOT 5006 C in serieller Abarbeitung zu testen und aufgetretene Fehlfunktionen dem Operator mitzuteilen. Es wird nur mit ungerader Parität gearbeitet. Die Auswahl des Aufzeichnungsverfahrens erfolgt nur über die Taste "NRZI" am jeweiligen Magnetbandgerät (siehe auch Beispiel unter Pkt. 5.1.6.4.).

5.1.2. Testbedingungen

5.1.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- SBA/SBE
- Konsolterminal
- Optionale Hardware: Eine oder mehrere der folgenden Konfigurationen:
  Kontroller ISOT 5006 C
  1 bis 4 Magnetbandgeräten CM 5306, CM 5308, CM 5309
  (einschließlich Datenträger)

5.1.2.2. Software

- Diagnose-Supervisor
- SVP 1800

Zum Ablauf dieses Programmes werden keine anderen vorrangig ablaufenden Programme benötigt.

5.1.3. Voraussetzungen

Dieses Programm verlangt die völlige Wirksamkeit aller Elemente im RVS (Speicher, CPU, SBA/SBE). Der Diagnose-Supervisor muß geladen sein und wird als Operatorinterface benutzt. Das Programm erfordert, daß jedes zu testende Laufwerk angeschlossen ist, Schreiben erlaubt (Schreibring auf Magnetbandspule) ist und daß die Geräte "on-line" geschaltet sind.

5.1.4. Bedienungshinweise

Muster der ATTACH-Folge:

```
DS> ATTACH DWA40 HUB DW0 3 4          ; DWA40 eingliedern
DS> ATTACH BM16 DW0 MEA0 772520 224 5
```

Start der Standardsektion:

```
DS> RUN PVMZA
```

Start der Qualifikationssektion:

```
DS> RUN PVMZA/SEC:QUALIFICATION
```

Start der Zuverlässigkeitssektion:

```
DS> RUN PVMZA/SEC:MEDIA
```

Start der Mehrfachlaufwerkssektion:

```
DS> RUN PVMZA/SEC:MULTI
```

Start der Acceptancesektion:

```
DS> RUN PVMZA/SEC:ACCEPTANCE
```

Start der Konversationsmodussektion:

```
DS> RUN PVMZA/SEC:CONVERSATION
```

5.1.5. Funktionsbeschreibung

5.1.5.1. Übersicht

Das Programm testet mit oder ohne Betriebssystem die Funktion der über den Kontroller ISOT 5006 C angeschlossenen MBG CM 5306, 5308, 5309. Die Betriebsarten NRZI oder PE werden nur über Tastendruck am Magnetbandgerät festgelegt.

5.1.5.2. Programmgröße

Das Programm benötigt einen Speicherplatz von 60 Kbyte.

5.1.5.3. Programmlaufzeit

Standardlaufzeit für ein Gerät CM 5306:

2,5 min für PE Aufzeichnungsart (1600 bpi)
3,5 min für NRZI Aufzeichnungsart (800 bpi)

5.1.5.4. Abarbeitungsbedingungen

Die DEFAULT (Standard)-Sektion schließt die Sektionen QUALIFICATION, DATA RELIABILITY und MULTI DRIVE ein. Die Tests sind mit den Sektionen identisch, deshalb werden diese nicht extra behandelt.

5.1.5.5. Flag-Verwendung

Mehrere vom Programm gelieferte Ereignisflags erlauben es dem Operator, die Kontrolle über das Programm zu erhöhen. All diese Flags (außer #3) können dynamisch gesetzt oder gelöscht werden. Das Flag #3 sollte vorrangig beim Aussenden eines Startkommandos gesetzt oder gelöscht werden.

EF #1 Wenn gesetzt, verbietet es Fehlerwiederholungen an allen Geräten für Soft-Errors.

EF #2 Wenn gesetzt, ist für alle Geräteereignisse Systemerrorlogging erlaubt.

EF #3 Wenn gesetzt, wird der Operator über den Wechsel der Anzahl der zu übertragenden Aufzeichnungen für den Test 4 (Mehrfachlaufwerkstest) gefragt. Standardwert ist 750.
Mit der Auswahl eines sehr hohen Wertes (z.B. 1000000) kann der Operator die Nutzung der gesamten Bandspule garantieren.
Zu berücksichtigen ist, daß dabei der Test einen kompletten Durchlauf von BOT bis EOT an allen ausgewählten Geräten ausführt, bevor der Programm-Endpaß erreicht wird.

5.1.5.6. Allgemeine Programmanforderungen

Parameter werden nur in der Sektion CONVERSATION angefordert (siehe Pkt. 5.1.5.) Die manuellen Handlungen "Testband mit Schreibring einlegen, Laden und Magnetbandgerät on-line schalten", sowohl die Auswahl der Aufzeichnungsart NRZI oder PE über Tastendruck am MBG sind notwendige Voraussetzungen für den Test des jeweiligen Magnetbandgerätes.

5.1.5.7. Fehlermitteilungen

5.1.5.7.1. Ausklammern bei off-line

Wenn ein Laufwerk während des Tests off-line geschalten wird, versucht das Diagnoseprogramm kontinuierlich den Rest der zu testenden Laufwerke zu prüfen. Das Laufwerk, welches durch off-line getrennt wurde, wird beim Start des nächsten Durchlaufes nochmal ausgewählt.

5.1.5.8. Fehlermeldung

Wird, nachdem eine I/O-Funktion ausgeführt wurde, durch das Programm im I/O-Statusblock ein Fehler entdeckt, wird die im Diagnosepuffer gespeicherte Information angezeigt. Nachfolgend ist ein Beispiel einer Fehlanzeige für ISOT 5006 C

mit CM 5306 für Test 2 aufgeführt. Die Bedeutung der Bits ist in Punkt 5.1.10. dargestellt.

```
******** PVMZA BM16 DATA RELIABILITY DIAGNOSTIC - 1.0 ********
  Pass 1, test 2, subtest 0, error 2, 6-jan-1988 14:55:43.13
  Device fatal error while testing MEAO: READ NEXT FUNCTION FAILED
  IN PE MODE PATTERN NUMBER 5
RECORD NUMBER =0
MMIOSB=020001A4

REGISTER        DATA            MNEMONICS

MTS:            00008000(X)     ; ILC
MTC:            0000E0C2(X)     ; ERR,DEN8,DEN5,CUR,INT-ENB,FUNCTION=1(X)
MTBRC:          000002OO(X)     ; BYTE CNT=00512(D)
MTCMA:          00008000(X)     ; ADRESSE A15-A00=8000(X)
MTD:            00000000(X)     ; BUFFER =00(X)
MTRD:           00000000(X)     ; READ-DATA=00(X)

******** End of Device fatal error number 2 ********

******** PVMZA BM16 DATA RELIABILITY DIAGNOSTIC - 1.0 ********
  Pass 1, test 2, subtest 0, error 5, 6-jan-1988 14:55:55.69
  Hard error while testing MEA0: DATA COMPARE ERROR AFTER READ NEXT (FORWARD)
  IN PE MODE PATTERN NUMBER 5
RECORD NUMBER =0

******** End of Hard error number 5 ********

ADDRESS         GOOD   BAD   XOR

00000800:       55     00    55    BIT6,BIT4,BIT2,BIT0
00000801:       55     00    55    BIT6,BIT4,BIT2,BIT0
00000802:       55     00    55    BIT6,BIT4,BIT2,BIT0
00000803:       55     00    55    BIT6,BIT4,BIT2,BIT0
   .            .      .     .     .
   .            .      .     .     .
   .            .      .     .     .
0000080F:       55     00    55    BIT6,BIT4,BIT2,BIT0

BYTES TRANSFERRED:512  BAD:512  BLOCK NO:85
```

5.1.5.9. Programmitteilungen

Durch Setzen des "TRACE"-Schalters mit SET TRACE wird jeder gestartete Test und der logische Name des zu testenden Gerätes ausgegeben. Am Ende eines Tests oder einer Testfolge erfolgt die "SUMMARY"-Ausschrift.
Bei mehreren Testdurchläufen erfolgt nach jedem Pass eine Ausgabe analog der "SUMMARY"-Ausschrift. Die "ENDPASS"- und die "SUMMARY"-Ausschriften haben für den "QUALIFICATIONSTEST" (Test 2) keine Bedeutung, da im Test 2 eine andere Fehlerauswertung erfolgt. Nur die Anzahl der entdeckten Fehler und die Anzahl der Durchläufe werden für den Test 2 erfaßt.

```
SUMMARY OF TOTAL ERRORS FOR _MEA0
WRITE SOFT ERRORS= 0
WRITE HARD ERRORS= 0
READ FORWARD SOFT ERRORS= 0
READ FORWARD HARD ERRORS= 0
TOTAL NUMBER OF BYTES TRANSFERED= 469039
.. End of run, 0 errors detected, pass count is 1
   time is 7-JAN-1988  13:55:59.08
DS>
```

5.1.5.10. Leistungsparameter des Testobjektes

entfällt;

5.1.5.11. Aktionen bei unerwarteten Fehlern

entfällt;

5.1.5.12. Testfolge bei mehreren Einheiten

Die Magnetbandgeräte werden entsprechend der Reihenfolge der Auswahl der Geräte getestet.

5.1.5.13. Einstellung für Ablauf unter AFT

entfällt;

5.1.5.14. Testsektionen

Dieses Programm enthält 5 Testsektionen. Jede von ihnen kann über einen Sektionsschalter in der Startkommandozeile für die Ausführung ausgewählt werden.

- QUALIFICATION (Test 2)

  Diese Sektion ist der Eignungstest für BM16 Magnetbandsysteme (MBK ISOT 5006 C mit MBG CM 5306/5308/5309). Der Test sendet alle legalen Kommandos zu allen ausgewählten Geräten aus und bestätigt die erfolgreiche Kommandobeendigung.

- MEDIA (Test 3)

  Diese Sektion ist ein Bandzuverlässigkeitstest mit Datenübertragungen zu und vom Band in einer Folge von kurzen herabzählenden Läufen bei Verwendung einer Variation der Aufzeichnungslänge und der Datenmuster. Die Geräte werden auf Basis eines zyklischen Multiplex-Verfahrens (round-robin) ausgewählt.

- MULTI DRIVE (Test 4)

Diese Sektion ist der Mehrfachlaufwerkstest. Sie ist ähnlich wie Sektion 3, außer daß der Geräte-"Service" asynchron ist, alle Bytezähler und Daten sind zufällig generiert und die Länge des genutzten Bandes ist vom Operator auswählbar.

- CONVERSATION MODE (Test 5, nur als Sektion aufrufbar)

  Diese Sektion ist der Konversationsmodustest, welcher dem Operator erlaubt, eigene Kommandofolgen, die zum Gerät gesendet werden, zu erzeugen.

- ACCEPTANCE (Test 3 und 4)

  In dieser Sektion erfolgt nacheinander die Abarbeitung des DATA-RELIABILITY(MEDIA)-Tests und des MULTI-DRIVE-Tests.

Das QUICK-Flag wird von diesen Tests nicht verwendet.

5.1.6. Testsektionenbeschreibungen

5.1.6.1. QUALIFICATIONS TEST
Eignungstest (Test 2)

Dieser Test sichert, daß die ISOT 1070C-Bandsysteme die folgenden Funktionen ausführen:

Rückspulen
Rücksetzen
Status lesen
Schreiben 1 Block (512 byte, Datenmuster 5)
Schreiben Bandmarke
1 Bandmarke Rücksetzen
1 Block Rücksetzen
Lesen 1 Block (512 byte, Datenvergleich mit Datenmuster 5)
1 Bandmarke Vorsetzen
Schreiben 1 Block (512 byte, Datenmuster 1)
1 Block Rücksetzen
Lesen 1 Block (512 byte, Datenvergleich mit Muster 1)
1 Block Rücksetzen
1 Block Vorsetzen

Beachte: Wenn sich ein Systemfehler ereignet, wird dieses Programm abgebrochen!

5.1.6.2. MEDIA TEST
Bandzuverlässigkeitstest (Test 3)

Wenn dieser Test gestartet ist, werden alle zu testenden Geräte zum Bandanfang (BOT) zurückgespult. Die auf jedem Band ablaufende Testfolge ist:

Schreiben, einen Block zurücksetzen, Lesen vorwärts

Nach jedem Lesen wird ein Datenvergleich ausgeführt. Jede Aufzeichnung ist 512 byte lang.
Das erste Wort jeder Aufzeichnung bestimmt die Block-Position auf dem Band. Dieser Testzyklus wird 5 mal mit dem zu testenden Gerät ausgeführt. Nachdem

dieser Zyklus beendet ist, wird das nächste Gerät getestet.
Wenn alle Geräte diesen Zyklus beendet haben, wird das nächste Datenmuster verwendet. Die Testfolge wird mit den Datenmustern 1-12 (siehe Punkt 5.1.8.) ausgeführt.
Wenn alle 60 Blöcke übertragen sind, startet das Programm seine nächste Testfolge, wobei es nach Rückspulen die Generierung zufälliger Parameter veranlaßt.
Dieser Test verwendet zuerst zufällige Bytezahlen (Grenzen 14 bis 5000) und er nutzt die gleiche Testfolge und die gleichen Datenmuster, wie sie im vorherigen Datentest verwendet wurden. 60 Blöcke werden zum/vom jedem ausgewählten Band übertragen. Nachdem alle 12 Datenmuster getestet wurden, wird dann ein zufälliges Datenmuster mit zufälligen Bytezählern verwendet.
Die Fehlerwiederholungslogik ist bei allen I/O-Request-Reihen erlaubt. Wenn das letzte Gerät getestet worden ist, wird eine Ende-Mitteilung ausgedruckt. Die Anzahl der übertragenen Byte wird angezeigt.

```
SUMMARY OF TOTAL ERRORS LOGICAL UNIT
WRITE SOFT ERRORS=
WRITE HARD ERRORS=
READ FORWARD SOFT ERRORS=
READ FORWARD HARD ERRORS=
TOTAL NUMBER OF BYTES TRANSFERED=
```

5.1.6.3. MULTI DRIVE TEST
Mehrfachlaufwerkstest (Test 4)

Dieser Test versucht eine maximale Wechselwirkung zwischen den zu testenden Geräten herzustellen. Das Programm wird durch Aussenden der QIO-Rufe zu allen zu testenden Geräten gestartet. Es wartet dann, bis es ein Laufwerk findet, das seine Funktion beendet hat. Dieses Laufwerk wird dann mit einer anderen Funktion beliefert.
Bei Verwendung dieser Testprozedur werden die zu testenden Geräte nicht in einer spezifischen Ordnung getestet. Das Laufwerk, welches zuerst seine Funktion beendet, veranlaßt das nächste Laufwerk, eine Funktion auszuführen.
Alle Geräte folgen der gleichen Testreihe. Die Anordnung der Testreihe ist folgende:

Rückspulen, Schreiben, Block Rücksetzen und Lesen

Die Rückspulfunktion wird nur bei Initialisierung dieses Tests ausgeführt. Datenvergleiche werden nach der Lesefunktion ausgeführt. Vor dem Aussenden eines Schreibkommandos zu dem zu testenden Gerät generiert das Programm eine Zufallsbytezahl und füllt einen Schreibpuffer mit Zufallsdaten. Das erfolgt für jedes zu testende Laufwerk. Es entstehen somit unterschiedliche Bytezähler und unterschiedliche Datenpuffer für jedes Laufwerk.
Nachdem jedes Laufwerk die gewünschte Anzahl Testfolgen abgeschlossen hat, wird es vom Test getrennt. Wenn das letzte Laufwerk seine letzte Testfolge beendet hat, ist das Testende erreicht.
Bei Verwendung des Standardwertes werden 750 Blöcke zum/vom Band übertragen.

Beachte: Das Programm erlaubt dem Operator die Anzahl der in diesem Test übertragenen Aufzeichnungen durch Setzen des Ereignisflags 3 zu wechseln. Siehe Pkt. 5.1.5.5. für mehr Details.

5.1.6.4. CONVERSATION MODE Konversationsmodustest (Test 5)

Der Konversationsmodustest erlaubt dem Operator eine Anzahl von Funktionen, die an dem zu testenden, gewünschten Gerät ausgeführt werden, zu spezifizieren. Bis zu 4 Laufwerke können bei diesem Test gleichzeitig ausgewählt werden.
Die Funktionsauswahl kann in einer gewünschten Reihenfolge durchgeführt werden. Dieses Programm prüft nicht die Funktionsfolge, um zu sehen, ob sie in einer logischen Aufeinanderfolge ist. Folglich können Fehler beim Laden einer unkorrekten Funktionsfolge erzeugt werden. Der Operator hat volle Verantwortlichkeit für das Laden der Funktionsfolgen.
Im Abschnitt 5.1.7. ist eine Liste von Funktionskodes enthalten, welche das Programm akzeptiert.
Für jedes ausgewählte Laufwerk werden Fragen gestellt. Jedes Laufwerk wird als ein Individuum betrachtet und hat seine eigene Testfolgetabelle, welche die auszuführenden Funktionen an dem spezifischen Laufwerk kontrolliert. Diese Reihenfolgetabelle wird durch das Programm aufgebaut, sobald der Operator die Fragen für das spezifische Laufwerk beantwortet.
Nachfolgend ist ein Beispiel von Fragen dargestellt, die für jedes beliebige Laufwerk gestellt werden kann. Dieses Beispiel ist für MEA0.

Beispiel:

Zuerst die CALLUP CONVERSATIONS SECTION in folgender Weise.

```
DS>START/SEC:CON
PLEASE ANSWER THE FOLLOWING QUESTIONS FOR MEA0:
NEW PARAMETERS WANTED?[(YES), NO, SHOW, HELP]
```

YES: Wenn der Operator "CARRIGE RETURN" (standardmäßig eine YES-Antwort) oder "YES" eingibt, fährt das Programm fort, Fragen für den Test des Gerätes zu stellen.

NO: Wenn der Operator "NO" eingibt, prüft das Programm die gegenwärtige Laufwerks-Folgetabelle. Ist die Information in dieser Tabelle gültig, verwendet das Programm die vorliegende Testfolge in der Tabelle zum Testen des Laufwerkes.
Es werden keine weiteren Fragen gestellt. Wenn die Information in der Folgetabelle ungültig ist, stellt das Programm die gleichen Fragen, die auf eine "YES" Antwort folgen.

SHOW: Antwortet der Operator "SHOW" , gibt das Programm auf Grundlage der Folgetabelle die Informationsbeschreibung, das Format und die Funktion aus, welche gegenwärtig für dieses Laufwerk existiert. Die obigen Fragen werden wiederholt, um zu sehen, ob der Operator die Folgetabelle ändern will.
Wenn die Information in dieser Tabelle defekt ist, fordert das Programm vom Operator Antwort auf die gestellten Fragen.

HELP: Wenn der Operator "HELP" eingibt, wird eine Mitteilung folgenden Inhaltes ausgegeben:
Gebe "Y" oder "N" auf die Frage nach Verwendung von Zufallsbytezahlen.
Zufallsmuster ist Muster 12.
Gebe eine Funktion pro Zeile ein. Ein Kode gefolgt von "!" veranlaßt nur eine Ausführung.
Beispiel Rückspulen: REW!
Nachfolgend sind die Funktionscodes aufgelistet:

NOP, DRCL, REW, WTM, SF, SFR, SR, SRR, WRT, RD, UNLD, END.

RANDOM BYTE COUNT [(Y) OR N]

Y: Wenn der Operator "Y" eingibt, generiert das Programm jedesmal, wenn die Funktionsfolge wiederholt wird, eine neue Zufalls-Bytezahl. Die generierte Bytezahl liegt zwischen 13 und 5000. Die nächste Frage wird weggelassen.

N: Wenn der Operator "N" eingibt, stellt das Programm die nachfolgenden Fragen.

SIZE OF BYTE COUNT [(5000)13-5000(D)]

Die eingegebene Bytezahl muß zwischen 13 und 5000 sein. Das Programm weist den Speicherbereich für den Puffer zu. Wenn nicht genug Speicher im System verfügbar ist, fragt das Programm den Operator, ob er eine kleinere Bytezahl eingeben will.

TYP PATTERN NUMBER [(12), 1-12(D)]

Der Operator kann jetzt ein Muster auswählen, welches er verwenden will. Die verfügbaren Muster sind im Abschnitt 5.1.8. aufgelistet. Wenn der Operator das Muster Nummer 12 (Zufallsdaten) auswählt, wird jedesmal wenn die Funktionsfolgetabelle wiederholt wird, ein neuer Datenpuffer generiert.

TYPE FUNCTIONES WANTED-TERMINATE WITH AN "END"

FUNC> Der Operator wählt nun die Funktionen aus, die er verwenden will. Nachdem die letzte Funktion eingegeben wurde, muß der Operator "END" eingeben, um die Eingabe zu beenden. Wenn ein unkorrekter Funktionskode eingegeben wurde, fragt das Programm, ob Hilfe gebraucht wird. Wenn der Operator "N" eingibt, fragt das Programm nach einer anderen Funktion durch Ausgabe von FUNC>. Der Operator kann dann kontinuierlich die Funktionen eingeben.
Wenn der Operator mit "Y" auf die HELP-Frage antwortet, gibt das Programm alle Funktionen aus und kehrt mit FUNC> zurück.

Damit sind die Fragen für das ausgesuchte Laufwerk komplett. Durch das Programm wird kein Test gestartet, bevor nicht für alle ausgewählten Laufwerke alle Fragen beantwortet sind. Ist dies erfolgt, sendet das Programm Funktionen zu all den Laufwerken, die durch ihre Folgetabellen ausgewählt wurden. Das Programm geht dann in eine Warteschleife und wartet bis ein Laufwerk seine QIO-Funktion beendet hat. Mit Hilfe der Laufwerksfolgetabelle wählt das Programm die nächste Funktion, sendet sie zu dem zu testenden Gerät und geht dann zurück, um auf ein anderes Laufwerk, welches seine QIO-Funktion beendet hat, zu warten.
Bei Verwendung dieser Testprozedur werden die zu testenden MBG nicht in einer spezifischen Ordnung getestet. Dasjenige Laufwerk, welches seine Funktion zuerst beendet, wird das Gerät sein, an das die nächste Funktion gesendet wird. Dieser Test wurde entwickelt, um auf diesem Weg ein maximales Zusammenwirken zwischen den ausgewählten Laufwerken auszuführen. Wenn alle Funktionen für ein spezifisches Gerät ausgeführt worden sind, kehrt das Programm zum Anfang dieser Laufwerksfolgetabelle zurück und wiederholt alle Funktionen noch einmal. Dieser Test hat keinen "END OF PASS" Abschluß. Er läuft solange, bis der Operator ein "^C" eingibt.

Beispiel: Neubeschreiben eines Bandes oder Wechsel des
Aufzeichnungsverfahrens von NRZI in PE oder umgekehrt

```
FUNC>REW!      ;Rückspulen nur einmalig(!)
FUNC>WRT       ;Schreiben eines Blockes
FUNC>WTM       ;Schreiben Bandmarke
FUNC>END       ;Endekennzeichen
```

Nach einmaligem Rückspulen erfolgt ständiges Schreiben von Blöcken und Bandmarken bis mit CTRL C unterbrochen wird.

5.1.7. Funktionskodes für den Konversations-Modustest

NOP: NO-OPERATION IO$_NOP
Führt keine Geräteoperation aus. Diese Funktion kann von dem Operator genutzt werden, um Kontroller und Geräteregister zu lesen ohne Ausführung einer aktuellen Funktion an dem ausgewählten Laufwerk.

DRCL: DRIVE CLEAR IO$_DRVCLR
Ausführung einer Laufwerks-Löschoperation.

REW: REWIND IO$_RECAL
Rückeinstellen der Bandposition, so daß die Reflexionsmarke am Bandanfang unter dem Lesekopf ist.

WTM: WRITE TAPEMARK IO$_WRITEMARK
Schreiben einer erweiterten Zwischenaufzeichnungs-Lücke (ungefähr 3inches) gefolgt von einer Bandmarke.

SF: SPACEFILE IO$_SPACEFILE
Zwischenraum nach einer Bandmarke bei Vorwärtssteuerung.

SFR: SPACEFILE REVERSE IO$_SPACEFILE
Zwischenraum nach einer Bandmarke bei Rückwärtsbewegung.

SR: SPACE RECORD IO$_SPACERECORD
Zwischenraum nach einer Aufzeichnung bei Vorwärtssteuerung.

SRR: SPACE RECORD REVERSE IO$_SPACERECORD
Zwischenraum nach einer Aufzeichnung bei Rückwärtssteuerung.

WRT: WRITE PHYSICAL BLOCK IO$_WRITEPBLK
Schreibe Daten von einem spezifischen Puffer zum Band beginnend bei der nächsten Blockposition in Vorwärtssteuerung.

RD: READ IO$_READPBLK
Lesen Daten vorwärts beginnend ab der nächsten Blockposition in den spezifizierten Speicherpuffer.

UNLD: UNLOAD IO$_UNLOAD
Rückspulen bis BOT und off-line am ausgewählten Laufwerk.

END: END - Kode wird vom Programm zum Abschließen der Funktionsfolgekette gebraucht.

5.1.8. Aufstellung der Datenmuster

Es folgt eine Liste der Datenmuster in der Reihenfolge, in welcher sie verwendet werden.

1. Alles "1" in allen Zeichen;
2. Alles "0" in allen Zeichen;
3. Eine "1" wandert von links nach rechts in einem Feld von Nullen;
4. Eine "0" wandert von links nach rechts in einem Feld von Einsen;
5. Wechselnde "1" und "0" in jedem Zeichen (5555H);
6. Wechselnde "0" und "1" in jedem Zeichen (AAAAH);
7. Die gleichen Daten wie 5. aber mit ein um das andere Zeichen vervollständigt (AAAAH) (5555H);
8. Eine "1" wandert durch ein Zeichen von alles "0" und alles "1" in wechselnden Zeichen (FF00H und wandernde 1);
9. Wechselnde Zeichen von alles "0" und alles "1" (FF);
10. Eine fließende "0" durch ein Zeichen von alles "1" und alles "0" in wechselnden Zeichen (FD);
11. Byte-Folgemuster 0, 0, -1, -1, 0, 0;
12. Zufalls-Daten-Muster;

5.1.9. Auflistung der Testsektionen

1. QUALIFICATIONS TEST
   Eignungstest
2. DATA RELIABILITY (MEDIA) TEST
   Bandzuverlässigkeitstest
3. MULTI DRIVE TEST
   Mehrfachlaufwerkstest
4. ACCEPTANCE TEST
   Annehmbarkeitstest
5. CONVERSATION MODE
   Konversationsmodustest

5.1.10. Register

MTS: Statusregister Bit 15 ... 0

Bit	Kurz-zeichen	Bedeutung
15	ILC	Ungültiges Kommando
14	EOF	Fileendemarke
13	CRE	(nicht benutzt)
12	PAE	Summenfehler
11	BGL	Busgewährung zu spät
10	EOT	Bandendemarke
9	RLE	Aufzeichnungslängenfehler
8	BTE	(nicht benutzt)
7	NXM	Kein existierender Speicher
6	SELR	Gerät ausgewählt
5	BOT	Bandanfangsmarke
4	7CH	(nicht benutzt)
3	SDWN	Anhalten Magnetband
2	WRL	Schreibschutz vorhanden (kein Schreibring)
1	RWS	Rückspulstatus
0	TUR	Magnetbandgerät bereit

MTC: Kommandoregister Bit 15 ... 0

Bit	Kurz-zeichen	Bedeutung
15	ERR	Fehler (MTS Bit 15...7)
14	DEN8	(keine Wirkung)
13	DEN5	(keine Wirkung)
12	PCLR	Programmtechnisches Rücksetzen
11	PEVN	(keine Wirkung)
10	NRZ	(NRZI - nicht benutzt)
9	USEL1	Geräteauswahl HW Bit
8	USEL0	Geräteauswahl NW Bit
7	CUR	Magnetbandkontroller bereit
6	INT ENB	Interrupterlaubnis
5	XBA17	Adreßbit 17
4	XBA16	Adreßbit 16
3-1	FUNCTION=	Funktionskode
0	GO	Startbit

MTBRC: Bytezählregister Bit 15 ... 0
Ausschrift erfolgt dezimal.

MTCMA: Adreßregister Bit 15 ... 0
18 Bit Register; Bit 16 und 17 in MTC;
Ausschrift erfolgt hexadezimal.

MTD: Datenpuffer Bit 7 ... 0 (15 ... 8)
8 Bit-Register;
Ausschrift erfolgt hexadezimal.

MTRD: Register für Längsprüfung Bit 15 ... 0

Bit	Kurz-zeichen	Bedeutung
15	TIM	Timer
14	CXC	=1=>Längsprüfung; =0=>zyklische Prüfung;
13	BTE-ERG	(nicht benutzt)
12	GSD	Verzögerung beim Anhalten
11-9		frei
8	PAR	Prüfbit der Längsprüfung
7-0	READ-DATA	Letztes gelesenes Datenbyte vom MBG; Muß nach Lesen eines gesamten Blockes "0" sein.

6. Diagnoseprogramme für Plattenspeichertechnik

6.1. FVRAA: Plattenzuverlässigkeitstest

6.1.1. Zusammenfassung

Das Diagnose-Programm FVRAA (Dok.-Nr. 1.57.700601.7) ist ein Programm der Ebene 2 (User- und Stand-alone-Modus). Es ist zum Nachweis der Arbeitszuverlässigkeit der Plattenspeicher PP05 (CM 5404), PP06 (CM 5416) und PM80 (K 5880) bestimmt.

6.1.2. Testbedingungen

6.1.2.1. Hardware

- RVS A 40 (mit Minimalkonfiguration)
- MSBUS-Adapter K 2816
- Optionale Hardware:
  Eine oder mehrere der folgenden Konfigurationen:
  RHA40 und 1 oder mehrere PP05
  RHA40 und 1 oder mehrere PP06
  RHA40 und 1 oder mehrere PM80

- Speichermedium

  Das Programm fordert auf jedem Gerät, das getestet werden soll, einen formatierten Plattenstapel. Die Platten sollen nicht eingegliedert sein. Der Plattenstapel muß ein Datenträgerkennzeichen ""SCRATCH"" oder ""DIAGNOSTIC"" haben. Wenn keine solche Kennzeichnung im Home-Block einer Platte steht, fordert das Programm den Nutzer auf, zu kontrollieren, ob die Fortsetzung des Testes auf das angesprochene Gerät richtig ist. Verneint der Nutzer diese Frage, wird der Test dieses Gerätes nicht durchgeführt.

- ERROR LOGGING
Im Normalfall ist die Error-Logging-Funktion nicht möglich. Wenn Plattenfehler im SYSTEM ERROR LOG ausgegeben werden sollen, muß der Nutzer das Ereignis-Flag 2 setzen.

6.1.2.2. Software

- Diagnose-Supervisor
- Driver PVQDB.EXE für PP05, PP06
  PVQDR.EXE für PM80

Ist der Plattenstapel unformatiert, muß erst das Plattenformatier-Programm zum Formatieren der Platte benutzt werden.

6.1.3. Voraussetzungen

Dieses Programm verlangt die volle Funktionsfähigkeit aller Rechnerbaugruppen. Der Diagnose-Supervisor PSSAA sollte geladen sein, weil er die Verbindung zwischen Nutzer und Programm darstellt.

6.1.4. Bedienungshinweise

Muster der ATTACH-Folge:

```
DS> ATTACH RHA40   HUB RH0 8 5      ;
DS> ATTACH PP06    RH0 DBA0         ;200 MB-WPS

DS> ATTACH RHA40   HUB RH1 9 5      ;
DS> ATTACH FM80    RH1 DRB0         ; Festplatte
```

6.1.5. Funktionsbeschreibung

6.1.5.1. Übersicht

Dieses Programm enthält 6 Testsektionen, von denen jede durch einen Sektionsschalter in der Startkommandozeile zur Abarbeitung ausgewählt werden kann. Diese Testsektionen sind:

QUALIFICATION	- Funktionstest aller Plattengeräte
SEEK.TIMING	- Positioniertest aller Plattengeräte
MEDIA.TEST	- Test des Speichermediums
MULTI-DRIVE	- Test der Parallelarbeit mehrerer Geräte
CONVERSATION MODE	- Wählbare Testfolge
NOCUSTOMER SECTION	- Erweiterter Mehrgerätetest

6.1.5.2. Programmgröße

Das Programm benötigt einen Speicherplatz von 256 Kbyte.

6.1.5.3. Programmlaufzeit

Normal: 34 min für 1 Gerät, 48 min für 2 Geräte (PP06)
Schnellauf: 10,5 min für 1 Gerät (PP06)

6.1.5.4. Abarbeitungsbedingungen

Die DEFAULT (Standard-)-Sektion schließt die Sektionen QUALIFICATION, SEEK TIMING und MULTI-DRIVE TEST ein. Die Tests sind mit den Sektionen identisch, deshalb werden diese nicht extra behandelt.

6.1.5.5. Flag - Verwendung

Bei gesetztem QUICK-Flag arbeitet das Programm einen Schnelldurchlauf von 10,5 Minuten ab (PP06).

Die EVENT-Flags 2 und 3 erlauben die Ausschrift von Einzelbitfehlern.

6.1.5.6. Allgemeine Programmanforderungen

keine

6.1.5.7. Fehlermitteilung

Beispielausschrift des Tests 1 QUALIFICATION (QA):

```
 ..Program: PVRAA DISK RELIABILITY, revision 1.0, 6 tests at 10:04:20.46.
Testing: _DBB3

_DBB3 IS AN PP05
VOLUME VERIFICATION DETECTS INVALID VOLUME NAME.
PACK NAME ON _DBB3 IS: &&&&&&&&&&
PACK NAME MUST BE ONE OF 'DIAGNOSTICS' OR 'SCRATCH'.
  DO YOU WISH TO OVERRIDE THE PROTECTION CHECK? [(NO), YES] Y
_DBB3 QA BEGUN AT 13-APR-1987 10:04:32.69

********  PVRAA   DISK RELIABILITY - 1.0  ********

 Pass 1, test 1, subtest 0, error 15, 13-APR-1987 10:04:33.59
 Hard error while testing DBB3:
```

```
FUNCTION INITIATION SUMMARY:

FUNCTION ATTEMPTED: WRITE DATA
BUFFER ADDRESS RANGE:    FROM: 00000391   TO 00000590
ATTEMPTED BYTE COUNT WAS: 512
STARTING DISK ADDRESS:
CYLINDER:  0  TRACK:  18 SECTOR:  0
FUNCTION ABORT SUMMARY
WRITE LOCK FOUND


MBACSR      00000020       ; ADAPTER CODE =20(X)

MBACR   :   00000004       ; IE
MBASR   :   00003080       ; DTCOMP,DTABT,MBEXC
MBAVAR  :   00000199       ; MAP POINTER=00(X),PAGE BYTE ADDRESS=199(X)
MBABCR  :   FE00FE08       ; MSBUS BYTE COUNT=FE00(X),ISB BYTE COUNT=FE08
MBAFMAP :   80000792       ; VALID,PFN=000792(X)
MBAPMAP :   00000000       ; PFN=000000(X)
RMCS1   :       0830       ; DVA,FUNCYION=WRITE DATA
RMDS    :       59C0       ; ERR,MOL,WRL,DPR,DRY,VV
RMER1   :       0800       ; WLE
RMMR    :       0000       ;
RMAS    :       0000       ;
RMDA    :       1201       ; TRACK=18(D),SECTOR=01(D)
RMDT    :       2017       ; MOH,DRIVE TYPE= PP05
RMLA    :       0740       ; SECTOR=29(D)
RMSN    :       4048       ; SERIAL NUMBER=4048(X)
RMOF    :       1000       ; FMT22
RMDC    :       0000       ; DESIRED CYLINDER=00000(D)
RMHR    :       EDFF       ; BIT15,BIT14,BIT13,BIT12,BIT11,BIT10,BIT9,BIT8,
                           ; BIT7,BIT6,BIT5,BIT4,BIT3,BIT2,BIT1,BIT0
RMMR2   :       13FF       ; CNT/CYL,BUS IN LINES=1FF(X)
RMER2   :       0000       ;
RMEC1   :       0836       ; BURST LOCATION=0836(X)V
RMEC2   :       0000       ; ERROR BURST=0000(X)
DRERL   :       0000       ;
********  End of Hard error number 15  ********
```

6.1.5.8. Programmitteilungen

Die Übersichtsausschrift enthält die Gesamtzahl der Übertragungen und die Gesamtzahl der Fehler, die während des Testes aufgetreten sind. Die Übersicht wird nur beim MEDIA TEST und beim MULTI DRIVE TEST ausgeschrieben.
Während der Ausführung der MEDIA- und MULTI DRIVE-Sektionen wird alle 5 Minuten eine Übersicht ausgegeben, wenn der "TRACE"-Schalter eingeschaltet ist. (Einschalten mit: SET TRACE). Am Ende jedes Durchlaufes wird die Übersicht auch gegeben, wenn dem Programm mehr als ein Durchlauf angewiesen worden ist. Die Fehlersummen werden über alle Durchläufe gebildet.

6.1.5.9. Leistungsparameter des Testobjektes

entfällt

6.1.5.10. Aktionen bei unerwarteten Fehlern

keine

6.1.5.11. Testfolge bei mehreren Einheiten

Reihenfolge der Arbeit mit den Geräten nach Reihenfolge der Auswahl der Geräte. Beim MULTI DRIVE-Test ist die Reihenfolge zufällig.

6.1.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.1.6. Testsektionsbeschreibungen

6.1.6.1. QUALIFIKATION (Standard) Gerätefunktionstest

Diese Testsektion testet einzeln jede Gerätefunktion, um zu sichern, daß die Geräte im Test alle Plattenspeicherkommandos ausführen. Dieser Test prüft nicht die nichtladbaren Kommandos oder einen Spindelstart. Nachdem alle Funktionen ohne Datentransfer getestet worden sind, werden eine Schreib-, Schreibtest- und Lesefolge zu den Testplattenadressen gesendet. Die FE-Zylinder werden bei Festplattengeräten mit benutzt.
Das Ziel der Lese/Schreibfolgen dieses Tests ist es, einen Sektor auf jedem Zylinder und jeder Spur zu erreichen, ohne den HOME-Block und die BADBLOCK-Datei, die sich auf der letzten Spur des letzten Zylinders befindet, zu zerstören und in weniger als einer Minute abzuschließen.

6.1.6.2. SEEK TIMING (Standard) Positioniertest

Dieser Test positioniert zwischen Zylinder 0 und allen folgenden binär gezählten Zylinderadressen bis zum maximalen Zylinder. Die Positionierzeit wird ermittelt und die Durchschnittszeit wird daraus berechnet. Die Ergebnisse werden in Tabellenform ausgegeben.

Anmerkung: Die Positionierzeiten sind nicht sehr genau, da SVP ein TIMESHARING-System ist und also die Geräte-Driver die QIO-, Stopp- und Startzeit in 10 Millisekunden-Abschnitte teilen.

6.1.6.3. MEDIA TEST
Schreib/Lese-Zuverlässigkeitstest

Dieser Test führt Schreiben und Schreibtest auf jedem Sektor der Platte aus und benutzt eine vollständige Spur für jede Operation. Ein Fehlerblock wird nicht ausgeschrieben, wenn er in der BADBLOCK-Datei steht, die auf der letzten Spur des letzten Zylinders steht. 5 Datenmuster werden auf jeden Sektor des ganzen Plattenstapels geschrieben. Jedes Muster besteht aus einem Quadwort das 64 mal in jeden Sektor geschrieben wird. Die folgenden Muster werden benutzt:

1. F00FF00FF00FF00F ; härtester Fall für den MSBUS-Adapter
2. EC6DEC6DEC6DEC6D ; härtester Fall für das Speichermedium
3. A5A5A5A5A5A5A5A5 ; alternierend 1 und 0
4. 123456789ABCDEF0 ;
5. FEDCBA9876543210 ;

Nachdem alle Sektoren mit den obigen Datenmustern beschrieben worden sind, geht der Test in einen Zufälligkeitsmodus über, wo zufällige Datenmuster auf zufällige Plattenadressen geschrieben werden.

6.1.6.4. MULTI-DRIVE TEST (Standard)
Mehrgerätetest

Diese Sektion testet bis zu 8 Geräte, indem zufällige Datenmuster zu zufälligen Plattenadressen aller Geräte in Parallelarbeit übertragen werden. Alle Übertragungen werden für die Geräte, die an separaten Kontrollern laufen parallel, für die, die am selben Kontroller laufen sequentiell, ausgeführt. Dieser Test arbeitet nur auf die FE-Zylinder der Festplattenspeicher. Die Funktionsfolge ist:

1. Gerät rücksetzen
2. Schreiben zufällige Daten
3. Schreibtest Daten
4. Lesen Daten
5. Datenvergleich

Anmerkung: Bei Fehler wird für das betreffende Gerät die Funktionsfolge mit Rücksetzen begonnen. Die Aktivitäten der anderen Geräte werden fortgesetzt.

6.1.6.5. CONVERSATION MODE
Modifizierbarer Servicetest

Dieser Test enthält einen Satz von Routinen, die dem Service-Ingenieur erlauben, selbst einfache Testroutinen mit geringer Schwierigkeit zu entwerfen und abzuarbeiten. BASIC-Funktionen wie "SEEK", "READ", "WRITE" usw. können benutzt werden. Es können bis zu 8 Geräte parallel bedient werden.
Um den CONVERSATION MODE zu starten, sind folgende Kommandos einzugeben (nach dem Laden des Diagnose-Supervisors):

niks werden durch Komma getrennt. Danach stellt das Programm folgende Fragen:

```
DO YOU WANT NEW PARAMETER RS:            ; mögliche Antworten sind YES,NO,
                                         ; SHOW oder HELP
ANSWER THE FOLOWING QUESTIONS FOR DBXX:
RANDOM BYTE COUNT?                       ; antworte YES oder NO (Standard: NO)
ENTER BYTE COUNT:                        ; Bereich ist 512 bis 64 Kbyte, wird
                                         ; nur abgefragt, wenn die erste Antwort
                                         ; NO,also "Kein zufälliger Bytezähler"
                                         ; eingegeben wurde
DATA PATTERN:                            ; Es gibt 12 verschiedene Datenmuster,
                                         ; die später in dieser Beschreibung
                                         ; stehen
RANDOM DISK ADRESS?                      ; Wird eine zufällige Plattenadresse
                                         ; gewünscht
ADRESS?                                  ;
ENTER FIRST CYLINDER:                    ; Diese Fragen erlauben es, eine Reihe
ENTER FIRST TRACK:                       ; von Plattenadressen zu testen
ENTER FIRST SECTOR:                      ; Die niedrigste Standardadresse ist
                                         ; jeweils 0
ENTER LAST CYLINDER:                     ; Die höchste Standardadresse ist der
ENTER LAST TRACK:                        ; letzte physikalische Zylinder, Spur
ENTER LAST SECTOR:                       ; oder Sektor

FUNC>:READ                               ; Das Programm ist nun bereit, die
FUNC>:WRITE                              ; Funktionen auszuführen.
FUNC>:SEEK                               ; Das Programm wird nach weiteren
FUNC>:END                                ; Funktionstests fragen, bis "END"
                                         ; eingegeben wird
```

An diesem Punkt stellt das Programm die Frage:

DO YOU WANT NEW PARAMETERS?

Bei der Antwort "YES" wird die Parametertabelle gelöscht und eine neue Routine kann zusammengestellt werden. Bei der Antwort "SHOW" wird die aktuelle Parametertabelle ausgeschrieben. Bei der Antwort "NO" beginnt das Programm mit der Ausführung der Routine.

Tabelle der Funktionen, die ausgeführt werden können:

RECALIBRATE	(Köpfe über Zylinder 0 positionieren)
SEEK	(Positionieren)
UNLOAD	
CLAER	(Rücksetzen Gerät)
NOP	(Keine Operation)
READ	(Lesen Daten)
READ HAEDER	(Lesen Kopfdaten)
WRITE	(Schreiben Daten)
WRITE HAEDER	(noch nicht vorhanden!)
WRITE CHECK	(Schreibtest)
WRITE CHECK HEAD	(noch nicht vorhanden)
SET/CYLINDER: N	(einstellen des gewünschten Zylinders n)
SET/TRACK: N	(" der " Spur n)
SET/SECTOR: N	(" des " Sektors n)
INCR/CYLINDER	(erhöhen des " Zylinders)
INCR/TRACK	(" der " Spur)
INCR/SECTOR	(" des " Sektors)
DECR/CYLINDER	(erniedrigen des " Zylinders)
DECR/TRACK	(" der " Spur)
DECR/SECTOR	(" des " Sektors)

Anmerkung: "SET-","INCR"- und "DECR"-Kommando werden nicht ausgeführt, wenn die Frage "RANDOM DISK ADRESS" mit "YES" beantwortet wurde. Funktionen nach denen ein "!" gesetzt wird, werden nur einmal ausgeführt.

Liste der nutzbaren Datenmuster:

1. alles Nullen
2. alles Einsen
3. A55A55A(X)
4. 5555555(X)
5. 8040201008040201(X) durchschieben "1" fallend;
6. 0102040810204080(X) durchschieben "1" steigend;
7. FEFDFBF7EFDFBF7F(X) " "0" "
8. 7FBFDFEFF7FBFDFE(X) " "0" fallend;
9. FCFCFCFCFCFCFCFC(X)
10. FF00FF00FF00FF00(X)
11. F0F0F0F0F0F0F0F0(X)
12. zufällig;

Beispiele:
1. Positionieren zwischen den Zylindern 0 und 128.

```
RANDOM BYTECOUNT? (N) Y OR N:    N
ENTER BYTECOUNT(512) 512-65536:  512
ENTER DATA PATTERN(12) 1-12:
RANDOM DISK ADRESS (N) Y OR N:   N
ENTER FIRST CYLINDER:            0
ENTER FIRST TRACK:               0
ENTER FIRST SECTOR:              0
ENTER LAST CYLINDER:             814
ENTER LAST TRACK:                18
ENTER LAST SECTOR:               21

FUNC: SET/CYLINDER:   0
FUNC: SEEK
```

```
FUNC: SET/CYLINDER:  128
FUNC: SEEK
FUNC: END
DO YOU WANT TO ENTER NEW PARAMETERS:  N
```

2. Schreiben, Lesen zufälliger Daten, zufälliger Bytezähler auf zufällige Plattenadressen.

```
RANDOM BYTECOUNT? (N) Y OR N:     Y
ENTER DATA PATTERN (12) 1-12:
RANDOM DISK ADRESS (N) Y OR N:    Y
ENTER FIRST CYLINDER:             0
ENTER FIRST TRACK:                0
ENTER FIRST SECTOR:               0
ENTER LAST CYLINDER:              814
ENTER LAST TRACK:                 18
ENTER LAST SECTOR:                21

FUNC: WRITE
FUNC: READ
FUNC: END
```

3. Beschreiben einer vollen Spur, Schreibtest und Erhöhen der Zylinderadresse.

```
RANDOM BYTECOUNT? (N) Y OR N:     N
ENTER BYTECOUNT (512) 512-65536: 11264
ENTER DATA PATTERN (12) 1-12:
RANDOM DISK ADRESS (N) Y OR N:    N
ENTER FIRST CYLINDER:             0
ENTER FIRST TRACK:                0
ENTER FIRST SECTOR:               0
ENTER LAST CYLINDER:              814
ENTER LAST TRACK:                 18
ENTER LAST SECTOR:                21

FUNC: WRITE
FUNC: WRITECHECK
FUNC: INCR/CYLINDER
FUNC: END
```

4. Einmaliges Rücksetzen, Schreiben einer vollen Spur auf Zylinder 0, Lesen einer vollen Spur auf Zylinder 800.

```
RANDOM BYTECOUNT? (N) Y OE N:     N
ENTER BYTECOUNT (512) 512-65536: 11264
ENTER DATA PATTERN (12) 1-12:
RANDOM DISK ADRESS (N) Y OR N:    N
ENTER FIRST CYLINDER:             0
ENTER FIRST TRACK:                0
ENTER FIRST SECTOR:               0
ENTER LAST CYLINDER:              814
ENTER LAST TRACK:                 18
ENTER LAST SECTOR:                21

FUNC: RECAL!
FUNC: SET/CYLINDER: 0
FUNC: WRITE
```

FUNC: SET/CYLINDER: 800
FUNC: READ
FUNC: END

6.1.6.6. NOCUSTOMER SECTION Erweiterter Mehrgerätetest

Dieser Test gleicht dem MULTI-DRIVE TEST, wie oben beschrieben, außer, daß bei Festplattenspeichern der gesamte physikalische Speicherbereich benutzt wird, ausgenommen die FE-Zylinder.

Warnung: Nutzerdaten werden zerstört!

6.2. PVRAC: Plattenformatierprogramm

6.2.1. Zusammenfassung

Das Diagnose-Programm PVRAC (Dok.-Nr. 1.57.700602.3) ist ein Programm der Ebene 2 (User- und Stand-alone-Modus). Es ist bestimmt zur Formatierung der PP05 (CM 5404)- und PP06 (CM 5416)-Wechselplatten.

6.2.2. Testbedingungen

6.2.2.1. Hardware

- RVS A 40 (mit Minimalkonfiguration)
- Optionale Hardware:
  Wechselplatten PP05 bzw. PP06

6.2.2.2. Software

- Diagnose-Supervisor mit oder ohne Betriebssystem SVP 1800 - Driver PVQDB.EXE

6.2.3. Voraussetzungen

- Vollständig getesteter Hardware-Kern;
- Gerätediagnose sollte fehlerfrei laufen

6.2.4. Bedienungshinweise (Details in HELP-Datei des Diagnose-Supervisors)

```
DS> ATTACH RHA40 HUB RH0 8 5
DS> ATTACH PP05 RH0 DBA0
DS> SELECT DBA0
DS> RUN PVRAC/SEC:FORMAT
```

6.2.5. Funktionsbeschreibung

6.2.5.1. Übersicht

Das Formatierprogramm besteht aus 7 Sektionen, bestehend aus einem oder mehreren gemeinsamen Teilen oder Tests. Z.B. besteht PACKINIT aus Test 1, 2,3 und 4. FORMAT besteht aus Test 2, 3 und 4.

6.2.5.2. Programmgröße

Das Programm erfordert mindestens 256 Kbyte Speicherplatz.

6.2.5.3. Programmlaufzeit

Ohne Betriebssystem in Minuten für FORMAT bzw. PACKINIT

Gerät	normal	schnell
PP05	25	12
PP06	65	27

6.2.5.4. Abarbeitungsbedingungen

Die Standardsektion ist eine Help-Sektion, die dem Nutzer die weitere Arbeit ermoeglicht. Die Sektionen setzen sich aus Tests zusammen, deren Einzelabarbeitung nicht sinnvoll ist. Deshalb sind diese nicht extra beschrieben.

6.2.5.5. Flag-Verwendung

6.2.5.5.1. QUICK

Das QUICK-Flag reduziert die Zahl der Datenmuster, die bei VERIFY benutzt werden, von 3 auf 1 (Die Datenmuster, die benutzt werden, sind alles Nullen, alles Einsen und härtestes Datenmuster. Ob QUICK gesetzt ist oder nicht, die Platte wird immer mit dem härtesten Datenmuster getestet).

6.2.5.5.2. EVENT-Flag 23

Das Ereignis-Flag 23 bewirkt, daß die Start-Plattenadresse zu jedem Schritt in einer Sektion ausgedruckt wird. In der FORMAT-Testsektion werden die Spuradressen formatiert. In der VERIFY-Testsektion werden die Spuradressen getestet.
Ausschrift wie folgt:

LBN = 45892(D) CYL = 695(D) TRK = 1(D) SEC = 0(D)

6.2.5.5.3. EVENT-Flag 22

Das Ereignis-Flag 22 bewirkt, daß die Start-Plattenadresse einmal pro 100 Zylinder ausgeschrieben wird. Dieses Flag unterdrückt Flag 23.
Ausschrift wie folgt:

LBN = 45892(D) CYL = 695(D) TRK = 1(D) SEC = 0(D)

6.2.5.5.4. EVENT-Flag 21

Ereignis-Flag 21 bewirkt, daß Fehlermeldungen in Kurzformat ausgegeben werden.

6.2.5.5.5. EVENT-Flag 20

Ereignis-Flag 20 wirkt nur im betriebssystemunabhängigen Modus. Dieses Flag bewirkt Einzelspurübertragungen im Oberflächentest.

6.2.5.5.6. EVENT-Flag 2

Wirkt nur im User-Modus.
Ereignis-Flag 2 erlaubt System-Fehlererfassung.

6.2.5.5.7. EVENT-Flag 1

Ereignis-Flag 1 erlaubt QIO-Fehlerwiederholungen. Im Normalbetrieb sollte dieses Flag nicht gesetzt sein.

6.2.5.6. Allgemeine Programmanforderungen

keine

6.2.5.7. Fehlerermittlung

Das Fehlerausschriftformat besteht aus 2 oder wahlweise 3 Teilen. Der erste Teil ist der zweizeilige Fehlerausschriftkopf. Der zweite Teil besteht aus dem Fehlerstatus dekodiert aus Register R0 als Hexadezimal-Fehlercode, dem Prozeß, der ihn feststellt (SVP, PSSAA oder PVRAC) und einer kurzen Definition. Der wahlweise dritte Teil schreibt die Geräteregister aus. Das Ausschreiben der Geräteregister wird vom IE2-Flag gesteuert. Das Ausschreiben der Fehlermitteilung wird durch das IE1-Flag gesteuert.

6.2.5.7.1. Gerätefehlerformat

Gerätefehler sind alle außer Datenfehlern, die im Gerät oder im Kanal, an dem das Gerät angeschlossen ist, vorkommen. Das Format enthält einen zweizeiligen Fehlerausschriftkopf, den Fehlerstatuscode, die Adapterregister, den QIO-Wiederholungszähler und Maximalzähler und die QIO-Start/Stop-Zeiten. Wenn Ereignisflag 21 gesetzt ist, werden die Adapterregister nicht ausgeschrieben, sondern nur der Plattenstatus, die Fehler- und die Plattenadreßregister.
Der QIO-Wiederholungszähler ist normal 0, wenn das Ereignisflag 1 Null ist. Der QIO-Maximalzähler ist die Maximalzahl der QIO-Wiederholungen bei Fehler, wenn Ereignisflag 1 gesetzt ist. QIO-Start/Stop-Zeiten sind die Start- und Stopzeiten des QIO, in dem ein Fehler vorkommt.

- Beispielfehlerausschrift (Langformat, EVENT-Flag 21 = 0)

```
******** PVRAC RVS DISK FORMATTER - 1.0 ********
 PASS 0  INITIALIZATION SECTION ERROR 9 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1:  INTERVENTION REQUIRED

STATUS RETURN  PROCESS          DEFINITION
_______________________________________________________________
00AC00C8 PVRAC MEDIUM OFFLINE

REGISTER     CONTENTS       BIT DEFINITIONS
_______________________________________________________________
MBACSR  :    00000020(X)    ; ADAPTER CODE=20(X)
MBACR   :    00000004(X)    ; IE
MBASR   :    00000000(X)    ;
MBAVAR  :    00000200(X)    ; MAP POINTER=01(X),PAGE BYTE
                            ; ADDRESS=0000(X)
MBABCR       00000000(X)    ; MSBUS BYTE COUNT=0000(X),
                            ; ISB BYTE COUNT=0000(X)
MBAFMAP :    00000000(X)    ; PAGE FRAME=000000(X)
MBAPMAP :    00000116(X)    ; VBIT,PAGE FRAME=000116(X)
RPCS1   :        0808(X)    ; DVA,FUNCTION=DRIVE CLEAR
RPDS    :        0980(X)    ; WRL,DPR,DRY
RPER1   :        0000(X)    ;
RPER2   :        0000(X)    ;
RPER3   :        0000(X)    ;
RPCC    :        0000(X)    ; CURRENT CYLINDER=00000(D)
RPOF    :        9000(X)    ; SGCH,FMT22
RPDA    :        0002(X)    ; TRACK=00(D),SECTOR=02(D)
RPLA    :        0540(X)    ; SECTOR=21(D)
RPDC    :        0000(X)    ; DESIRED CYLINDER=00000(D)
RPMR    :        1000(X)    ; ZDI
RPDT    :        2011(X)    ; MOH,DRIVE TYPE=PP05
RPAS    :        0000(X)    ;
RPSN    :        0167(X)    ; SERIAL NUMBER=0167(X)
RPEC1   :        0000(X)    ; BURST LOCATION=0000(X)
RPEC2   :        0000(X)    ; ERROR BURST=0000(X)

FINAL ERROR COUNT=   0
MAX   ERROR COUNT=   8

QIO STOP  TIME= 11:54:38.02
QIO START TIME= 11:54:38.02

 ..ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

- Beispielfehlerausschrift (Kurzformat, EVENT-Flag 21 = 1)

```
********  PVRAC RVS DISK FORMATTER - 1.0  ********
 PASS 0  INITIALIZATION SECTION  ERROR 9 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1:  INTERVENTION REQUIRED

STATUS RETURN    PROCESS          DEFINITION
_______________________________________________________________
00AC00C8 PVRAC   MEDIUM OFFLINE
```

```
REGISTER     CONTENTS        BIT DEFINITIONS
-----------------------------------------------------------
RPCS1   :      0808(X)     ; DVA,FUNCTION=DRIVE CLEAR
RPDS    :      0980(X)     ; WRL,DPR,DRY
RPER1   :      0000(X)     ;
RPER2   :      0000(X)     ;
RPER3   :      0000(X)     ;
RPCC    :      0000(X)     ; CURRENT CYLINDER=00000(D)
RPOF    :      9000(X)     ; SGCH,FMT22
RPDA    :      0002(X)     ; TRACK=00(D),SECTOR=02(D)


FINAL ERROR COUNT=   0
MAX   ERROR COUNT=   8

QIO STOP  TIME= 11:54:38.02
QIO START TIME= 11:54:38.02

 ..ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

6.2.5.7.2. Systemservicefehlerformat

Ein Systemservicefehler ist ein Fehler, der entweder vom Diagnose-Supervisor oder dem Betriebssystem festgestellt wurde, während das Formatierprogramm gerade eine Serviceroutine anfordert (z.B. $DS_GETBUF)

- Beispiel einer Systemservicefehlermitteilung

```
******** PVRAC RVS DISK FORMATTER - 1.0  ********
 PASS 0  INITIALIZATION SECTION  ERROR 5 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1:   FAILED TO OBTAIN BUFFER MEMORY

STATUS RETURN    PROCESS           DEFINITION
-----------------------------------------------------------
00000244 VMS     VIRTUAL     ADDRESS SPACE FULL

 ..ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

6.2.5.7.3. READALL Datenfehlerausschrift

- Beispiel einer READALL Datenfehlerausschrift

```
**********  READALL DATA ERROR REPORT  **********

TRACK= 0(D)

    TIMES ERROR LISTED IN BAD
  CYLINDER  SECTOR  RETRIED  CODE SECTOR  FILE
-----------------------------------------------------------
     95(D)     11(D)    10(D)  UNECC          NO
    112(D)      4(D)     5(D)  ECC            NO
    157(D)     18(D)    10(D)  BSE            YES

TRACK= 4(D)
```

```
    TIMES ERROR LISTED IN BAD
  CYLINDER  SECTOR  RETRIED  CODE SECTOR  FILE
-------------------------------------------------------
    95(D)      3(D)    10(D)  UNECC         NO
   295(D)     21(D)    10(D)  HCRC          NO

(THE FOLLOWING IS FROM THE SUMMARY REPORT)

FOR PHYSICAL DEVICE _DBA0 DRIVE TYPE= PP05
PACK IS LABELED 'SCRATCH      '
PACK SERIAL NUMBER=       1234(D)
NUMBER OF BAD SECTORS FOUND DURING READALL=        5(D)
```

- Definitionen von Codes der READALL-Sektion

Code	Definition
ECC	Korrigierbarer Datenfehler
UNECC	Unkorrigierbarer Datenfehler
BSE	BAD-Sektorfehler (Bit 15 oder Bit 14 des Kopfwortes ist gelöscht)
HCRC	Kopf-CRC-Fehler
HCE	Kopf-Vergleichsfehler
FER	Formatfehler
OPI	Operation unvollständig
DTE	Gerätezeitfehler
SBE	Synchronbytefehler

- Wiederholzählereinstellung

Wenn ein Datenfehler auftritt, wird der Fehlersektor wiederholt gelesen, bis sich der Fehler ändert, der Sektor lesefehlerfrei ist oder bis der maximale Wiederholzähler 10 erreicht ist. Der Zähler zeigt, wieviele Male der Fehler auftrat. Ein Zählerstand 0 hat besondere Bedeutung. Wenn während des Lesens eines Sektors der Fehler sich ändert (z.B. ein unkorrigierbarer Fehler wird zu einem korrigierbaren) werden beide Fehler registriert, doch der letzte Fehler erhält einen Zähler von 0.

6.2.5.8. Programmitteilungen

siehe Flag-Verwendung

6.2.5.9. Leistungsparameter des Testobjektes

entfällt

6.2.5.10. Aktionen bei unerwarteten Fehlern

Neubeginn der Programmabarbeitung

6.2.5.11. Testfolge bei mehreren Einheiten

Die Geräte werden in der Reihenfolge, in der sie ausgewählt wurden, abgearbeitet.

6.2.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.2.5.13. Testsektionen

PACKINIT	(Formatieren mit Erstellen BADBLOCK-Datei)
FORMAT	(Formatieren mit Korrektur BADBLOCK-Datei)
VERIFY	(Stapel prüfen mit Korrektur BADBLOCK-Datei)
READALL	(Datentest)
FLAGBAD	(Erstellen BADBLOCK-Datei von Hand)
PXFORMAT	(reserviert)
INCLUDE	(Ausgabe der BADBLOCK-Datei)
DEFAULT	(Standardsektion ist die HELP-Sektion)
HELP	(Standardsektion, nur zur Information)
FIXDTR	(reserviert)
SCANNER	(reserviert)
FMTPP07	(reserviert)
MODIFYTDR	(reserviert)

6.2.6. Testsektionenbeschreibungen

6.2.6.1. PACKINIT

Diese Sektion formatiert und schreibt eine Null-Eingangs-BADBLOCK-Datei auf alle Sektoren der letzten Spur. Die Sektion liest dann und macht die BADBLOCK-Datei gültig. Das bedeutet, daß die Datei an das gültige Format angepaßt ist, und daß alle physikalischen Plattenadressen innerhalb der Grenzwerte des Gerätes sind (Wenn die Datei nicht gelesen werden kann oder unkorrekt ist, wird das Programm abgebrochen).
Nach dem erfolgreichen Lesen der BADBLOCK-Datei wird der Stapel spurweise formatiert. Nach dem Abschluß des Formatiervorganges wird ein Schreib-Lese-Zyklus auf jeden Sektor des Plattenstapels durchgeführt. Während der Oberflächenprüfung wird jeder schlechte Sektor entdeckt, gekennzeichnet und der BADBLOCK-Datei hinzugefügt. Der HOMEBLOCK wird in einem Pseudo-ODS2-Format mit einem Kennungsnamen "SCRATCH" geschrieben (siehe Abschnitt 32.5.13.3.).
(Der Term "Pseudo ODS2" ist hier definiert als ein HOMEBLOCK mit einer 200(HEX), geschrieben auf HM2$W_STRUCLEV, dem ASCII-Namen "SCRATCH", geschrieben auf HM2$T_VOLNAME und allen anderen Plätzen auf Null gesetzt. Dies erlaubt Plattentestprogrammen, einen gültigen Kennungsnamen zu lesen, aber meldet ungültige HOMEBLOCK-Erscheinungen zum SVP.)

Diese Sektion sollte nicht gestartet werden, ohne daß der Nutzer überprüft hat, daß das Plattensubsystem fehlerfrei läuft und daß die BADBLOCK-Datei wirklich fehlt oder zerstört ist.
Diese Sektion ist für alle unbekannten Plattenstapel zu benutzen und für solche, auf die vorher keine BADBLOCK-Datei geschrieben wurde.
Achtung! Die erstellte BADBLOCK-Datei wird nicht eher auf die Platte geschrieben, bis die VERIFY-Sektion beendet ist.

- Beispielausschrift von PACKINIT

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:PACKINIT
 .. PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0

DISK TYPE= PP06 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2    PACK LABEL= 'RVSSVP
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE? (N), Y OR N : Y

************ WARNING ************
BAD SECTOR FILE WILL BE REFORMATTED

DO YOU WISH TO CONTINUE? (N), Y OR N : Y

FOR PHYSICAL DEVICE _DBA0
ENTER PACK SERIAL NUMBER (1), 1-2147383647(D) : 12345


CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE
FORMAT STARTET      AT 12:31:47.60 FOR DEVICE _DBA0
FORMAT COMPLETED    AT 12:35:23.37 FOR DEVICE _DBA0
VERIFY STARTET      AT 12:35:25.08 FOR DEVICE _DBA0
VERIFY COMPLETED    AT 12:52:09.54 FOR DEVICE _DBA0
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP06
PACK WAS LABELED 'RVSSVP       '
PACK SERIAL NUMBER=       12345(D)
NUMBER OF BAD SECTORS ADDED TO BAD SECTOR FILE=   0(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

6.2.6.2. FORMAT

Diese Sektion liest die BADBLOCK-Datei und macht sie gültig (Wenn die Datei nicht gelesen werden kann oder falsch ist, wird das Programm abgebrochen). Nach erfolgreichem Lesen der BADBLOCK-Datei wird der Plattenstapel spurweise formatiert. Nach Beendigung des Formatiervorganges wird eine Oberflächenprüfung durchgeführt, und jeder entdeckte schlechte Sektor wird markiert und der BADBLOCK-Datei hinzugefügt. Der HOMEBLOCK wird in einem Pseudo ODS2-Format mit dem Kennungsnamen "SCRATCH" geschrieben.

Die BADBLOCK-Datei wird erst auf die Platte geschrieben, wenn die VERIFY-Sektion beendet ist.

- Beispielausschrift von FORMAT

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:FORMAT
 .. PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0

DISK TYPE= PP05 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2     PACK LABEL= 'RVSSVP       '
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE?  (N), Y OR N : Y

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=      12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE
FORMAT STARTET      AT 12:31:47.60 FOR DEVICE _DBA0
FORMAT COMPLETED    AT 12:35:23.37 FOR DEVICE _DBA0
VERIFY STARTET      AT 12:35:25.08 FOR DEVICE _DBA0
VERIFY COMPLETED    AT 12:52:09.54 FOR DEVICE _DBA0
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP05
PACK WAS LABELED 'RVSSVP      '
PACK SERIAL NUMBER=      12345(D)
NUMBER OF BAD SECTORS ADDED TO BAD SECTOR FILE=   0(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

6.2.6.3. VERIFY

Diese Sektion liest die BADBLOCK-Datei und macht sie gültig (Wenn die Datei nicht gelesen werden kann oder falsch ist, wird das Programm abgebrochen.) Nach erfolgreichem Lesen der BADBLOCK-Datei wird die Oberfläche des Stapels geprüft und jeder entdeckte fehlerhafte Sektor wird markiert und der BADBLOCK-Datei hinzugefügt. Der HOMEBLOCK wird in einem Pseudo ODS2-Format mit dem Kennungsnamen "SCRATCH" geschrieben.

Die erstellte BADBLOCK-Datei wird erst auf die Platte geschrieben, wenn die VERIFY-Sektion beendet ist.

- Beispielausschrift für VERIFY

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:VERIFY
 .. PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0
```

```
DISK TYPE= PP05 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2    PACK LABEL= 'RVSSVP
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE?  (N), Y OR N : Y

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE
VERIFY STARTET       AT 12:35:25.08 FOR DEVICE _DBA0
VERIFY COMPLETED     AT 12:52:09.54 FOR DEVICE _DBA0
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP05
PACK WAS LABELED 'RVSSVP       '
PACK SERIAL NUMBER=       12345(D)
NUMBER OF BAD SECTORS ADDED TO BAD SECTOR FILE=    0(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

6.2.6.4. READALL

Diese Sektion liest jeden Sektor eines Stapels und schreibt jeden gefundenen Datenfehler in eine Kopf-Fehlermitteilung ein. Bei einem guten Plattengerät kann diese Sektion zur Suche von Datenfehlern eines Plattenpaketes benutzt werden. Umgekehrt kann diese Sektion mit Hilfe eines bekanntermaßen guten Stapels mit bekannten Fehlern zum Prüfen der Lesefähigkeit eines Plattengerätes benutzt werden.

- Beispielausschrift von READALL

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:READALL
 ...PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0

DISK TYPE= PP05 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2    PACK LABEL= 'RVSSVP
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE?  (N), Y OR N : Y

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE
READALL STARTET      AT 12:35:25.08 FOR DEVICE _DBA0
READALL COMPLETED    AT 12:39:09.54 FOR DEVICE _DBA0
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP05
PACK WAS LABELED 'RVSSVP       '
PACK SERIAL NUMBER=       12345(D)
NUMBER OF BAD SECTORS FOUND DURING READALL=      0(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

6.2.6.5. FLAGBAD

Diese Sektion erlaubt dem Nutzer, die BADBLOCK-Dateien von Hand zu erstellen. Diese Sektion fordert den Nutzer zur Eingabe der Adresse des BADBLOCK's auf. Nachdem der Nutzer alle BADBLOCK's, die er kennzeichnen will, ausgewählt hat, schreibt das Programm die BADBLOCK-Datei zurück und löscht das BADBLOCK-Flag im Kopfwort des BADBLOCK's (bei PP05 und PP06 wird nur die BADBLOCK-Datei zurückgeschrieben). Das Programm schreibt dann den HOMEBLOCK zurück, um die Datei-Directory zu zerstören. Dies wird getan, um zu erreichen, daß keine Datei eine Lücke aufweist. Das Programm liest dann die BADBLOCK-Datei und schreibt deren Inhalt aus.

> Diese Sektion sollte nicht benutzt werden, wenn der Nutzer nicht mit der Struktur der BADBLOCK-Datei und der Bedeutung von deren Erstellung vertraut ist. Nutzerdaten auf der Platte werden nicht erhalten. Der HOMEBLOCK wird in einem Pseudo-ODS2-Format mit dem Kennungsnamen "SKRATCH" auf die Platte geschrieben.

- Beispielausschrift von FLAGBAD

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:FLAGBAD
 .. PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0

DISK TYPE= PP06 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2   PACK LABEL= 'RVSSVP
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE?  (N), Y OR N : Y

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=        12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE

************ WARNING ************
BAD SECTOR FILE WILL BE MODIFIED BY THIS SECTION

DO YOU WISH TO CONTINUE?  (N), Y OR N : Y
SELECT INPUT TYPE (LOGICAL), PHYSICAL : PHYSICAL
CYLINDER  (1), 0-814(D) : 455
TRACK  (0), 0-2(D) : 1
SECTOR  (0), 0-21(D) : 19
DO YOU WISH TO FLAG MORE BAD BLOCKS?  (N), Y OR N : N
BAD SECTOR FILE UPDATED BY      1(D) ENTRIES

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
LOGICAL BLOCK      CYLINDER      TRACK       SECTOR
____________________________________________________
     30071(D)          455(D)       1(D)        19(D)
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP06
PACK WAS LABELED 'RVSSVP       '
```

```
PACK SERIAL NUMBER=       12345(D)
NUMBER OF BAD SECTORS ADDED TO BAD SECTOR FILE=       1(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

- Weitere Beispielausschrift von FLAGBAD

```
DS> SELECT DBA0
DS> RUN PVRAC/SECTION:FLAGBAD
 .. PROGRAM: PVRAC RVS DISK FORMATTER, REV 1.0, 7 TESTS.
TESTING: _DBA0

DISK TYPE= PP06 FOR PHYSICAL DEVICE _DBA0
PACK STRUCTURE= ODS2   PACK LABEL= 'RVSSVP
PACK IS NOT LABELED 'SCRATCH'

DO YOU WISH TO CONTINUE? (N), Y OR N : Y

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
NO BAD SECTORS IN FILE

************ WARNING ************
BAD SECTOR FILE WILL BE MODIFIED BY THIS SECTION!

DO YOU WISH TO CONTINUE? (N), Y OR N : Y
SELECT INPUT TYPE (LOGICAL), PHYSICAL : LOGICAL
LOGICAL  (2), 2-53767(D) : 30071
DO YOU WISH TO FLAG MORE BAD BLOCKS? (N), Y OR N : N
BAD SECTOR FILE UPDATED BY      1(D) ENTRIES

CONTENTS OF BAD SECTOR FILE
PACK SERIAL NUMBER=       12345(D) FOR DEVICE _DBA0
LOGICAL BLOCK      CYLINDER       TRACK       SECTOR
-----------------------------------------------------
     30071(D)        455(D)        1(D)        19(D)
FOR PHYSICAL DEVICE _DBA0 DISK TYPE=PP06
PACK WAS LABELED 'RVSSVP       '
PACK SERIAL NUMBER=       12345(D)
NUMBER OF BAD SECTORS ADDED TO BAD SECTOR FILE=       1(D)
 ..END OF RUN. 0 ERRORS DETECTED. PASS COUNT:1. TIME:29-JAN-1987 12:53:00.33
```

6.2.6.6. PXFORMAT

reserviert

6.2.6.7. INCLUDE

Diese Sektion gibt die BADBLOCK-Datei, die Plattenstapelseriennummer, die physische Bezeichnung des Plattengerätes, die Plattenstapelkennzeichnung und die Zahl der Defektsektoren aus.

6.2.6.8. HELP

Dies ist die Standard-Sektion. Die Sektionsnamen sind identisch mit ihrer Funktion. Nutzbare Ereignis-Flags sind beschrieben.

6.2.6.9. FIXTDR, SCANNER, FMTPP07, MODIFYTDR

reserviert

6.2.7. Auflistung der Tests

Nur zur Information! Eine Einzelabarbeitung der Tests ist nicht sinnvoll.
Test 01: Initialisieren BADSECTOR-Datei
Test 02: Anzeigen BADSECTOR-Datei
Test 03: Stapelformatieren
Test 04: Oberflächenprüfung
Test 05: Lesetest
Test 06: Erstellen der BADSECTOR-Datei von Hand
Test 07: reserviert
Test 08: reserviert
Test 09: reserviert
Test 10: reserviert
Test 11: Nutzungshilfe

6.3. PVRAD: Plattenspeicherfunktionstest

6.3.1. Zusammenfassung

Das Plattendiagnoseprogramm PVRAD (Dok.-Nr. 1.57.700603.1) ist ein Programm der Ebene 2 (User- und Stand-alone-Modus) und führt den zerstörungsfreien ONLINE-Funktionstest einer Anzahl von Plattensubsystemen durch. Da der Test wirklich zerstörungsfrei ist, kann man dieses Programm nicht nur für freie Arbeitsplatten (SCRATCH) benutzen. Dieses Programm hat auch die Fähigkeit, Nutzergeräte zu testen, die als Systemplatten unter SVP 1800 eingegliedert sind. Bis zu 8 Geräten können für den Test ausgewählt werden.

6.3.2. Testbedingungen

6.3.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- MSBUS-Adapter K 2816
- Testgeräte: PP05, PP06, PM80

6.3.2.2. Software

- Diagnose-Supervisor mit oder ohne SVP 1800
- Driver: PVQDB.EXE für PP05, PP06
  PVQDR.EXE für PM80

6.3.3. Voraussetzungen

- Vollständig getestete Hardware;

6.3.4. Bedienungshinweise

6.3.4.1. Anschließen

```
DS> ATTACH DWA40 HUB RH0 8 5
DS> ATTACH PP06 RH0 DBA0
DS> LOAD PVRAD
DS> SELECT DBA0
DS> START
 ... PROGRAM: ...
TESTING: _DBA0
```

6.3.4.2. Anschließen von Nutzerplatten

```
 ..PROGRAM: PVRAD LEVEL 2 DISK FUNKTIONAL DIAGNOSTIC REV 1.0, 30 TESTS
TESTING: _DBA0
DS> LOAD PVRAD                     ! Programm muß erst geladen werden
```

```
DS> ATTACH DWA40 HUB RH0 8 5
DS> ATTACH PUBDISK RH0 DBA0          ; Systemplatte
DS> SELECT DBA0
DS> START
```

6.3.5. Funktionsbeschreibung

6.3.5.1. Übersicht

PVRAD ist ein Funktionsdiagnoseprogramm der Ebene 2 und arbeitet unter dem Diagnose-Supervisor. Das Programm arbeitet ohne die Nutzerdaten auf den Platten zu zerstören. Es ist keine spezielle Arbeitsplatte notwendig.
Der Test wird aus 30 Einzeltests zusammengestellt, wobei jede Testgruppe mit dem zuerst ausgewählten Gerät arbeitet und dann mit dem nächsten ausgewählten Gerät fortgefahren wird.

6.3.5.2. Programmgröße

PVRAD benötigt 43,5 Kbyte Speicherplatz oder 85 Seiten. Viele der Programmdatenpuffer befinden sich innerhalb des Programms, doch werden vom Programm zusätzlich 50 Seiten dynamischer Speicher für größere DMA-Transfers gefordert.

6.3.5.3. Programmlaufzeit

Gerät	Zeit/min	schnell
PP05	7	
PP06	7	0,5
PM80	6	

6.3.5.4. Abarbeitungsbedingungen

Dieses Programm ist nur für digital strukturierte Medien geeignet. Nur die spezifizierten Plattengeräte dürfen für den Test ausgewählt werden, und die Platte muß nicht privat zugewiesen werden.

Datenschutz:
PVRAD wurde unter der Vorgabe entwickelt, Nutzerdaten geschützt zu halten. Das Folgende ist eine kurze Beschreibung der Hilfsmittel, mit denen dies erreicht wird.
Das Wichtigste ist, daß das Programm so wenig wie möglich Bedieneingriffe fordert. Es gibt keine Bedienfehler, die zu einem Datenverlust führen könnten. Bei harten Gerätefehlern wird das Programm abgebrochen. Alle Schreiboperationen

werden, wenn das Gerät so ausgerüstet ist, auf die FE-Zylinder oder auf eine zweite Kopie der 18-bit-Modus-BADBLOCK-Datei ausgeführt, die von RVS/SVP nicht benutzt wird. Sollte es der Nutzer für nötig halten, existiert die NOWRITE-Sektion zum Schutz der allgemein nutzbaren Platten. Platten, die nicht von jedem benutzt werden können, werden automatisch vom Programm vor Schreiben geschützt.

6.3.5.5. Flagverwendung

6.3.5.5.1. Ereignisflag 21

Ereignisflag 21 bewirkt, daß Fehlermitteilungen in Kurzformat ausgegeben werden.

6.3.5.5.2. Ereignisflag 2

Ereignisflag 2 erlaubt Systemfehlerprotokollierung. Wenn Fehlerprotokollierung erlaubt ist, werden alle Tests, die Gerätefehler provozieren, ausgelassen. Dieses Flag ist nur im Nutzermodus effektiv.

6.3.5.5.3. Ereignisflag 1

Ereignisflag 1 erlaubt QIO-Fehlerwiederholungen. Bei normaler Nutzung sollte dieses Flag nicht gesetzt sein.

6.3.5.6. Allgemeine Programmanforderungen

keine

6.3.5.7. Fehlermitteilung

Das Fehlerausschriftformat besteht aus 2 oder wahlweise 3 Teilen. Der erste Teil ist der zweizeilige Fehlerkopf. Der zweite Teil besteht aus dem Fehlerstatuscode des Registers R0 (HEX), dem Prozeß, der ihn meldet (RVS, PSSAA oder PVRAD) und einer kurzen Definition. Im wahlweise dritten Teil werden die Geräteregister ausgegeben. Die Ausgabe der Geräteregister wird von dem IE3-Flag gesteuert. Die Ausgabe des Statuscodes wird von dem IE2-Flag gesteuert. Die Ausgabe der Fehlermitteilung wird von dem IE1-Flag gesteuert.

6.3.5.7.1. Gerätefehlerformat

Ein Gerätefehler ist jeder Nichtdatenfehler, der im Gerät oder dem Kanal, an dem das Gerät angegliedert ist, auftritt. Das Format besteht aus einem zweizeiligen Fehlerkopf, dem Fehlerstatuscode, den Adapterregistern, den Geräteregistern, dem QIO-Wiederhol- und -Maximalzähler und den QIO-Start/Stopp-Zeiten.
Ist Ereignisflag 21 gesetzt, werden nicht die Adapterregister, sondern nur der Plattenstatus, Fehler- und Plattenadreßregister ausgeschrieben.
Der QIO-Wiederholzähler ist 0, wenn Ereignisflag 1 gelöscht ist. Der QIO-

```
******** PVRAD LEVEL 2 DISK FUNCTIONAL DIAGNOSTIC - 1.0 ********
 PASS 0 INITIALIZATION SECTION ERROR 9 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1:  INTERVENTION REQUIRED

STATUS RETURN       PROCESS       DEFINITION
-----------------------------------------------------------------
00AC00C8 PVRAD      MEDIUM OFFLINE

REGISTER     CONTENTS          BIT DEFINITIONS
-----------------------------------------------------------------
MBACSR  :    00000020(X) ; ADAPTER CODE=20(X)
MBACR   :    00000004(X) ; IE
MBASR   :    00000000(X) ;
MBAVAR  :    00000200(X) ; MAP POINTER=01(X),
                         ; PAGE BYTE ADDRESS=0000(X)
MBABCR  :    00000000(X) ; MSBUS BYTE COUNT=0000(X),
                         ; ISB BYTE COUNT=0000(X)
MBAFMAP :    00000000(X) ; PAGE FRAME=000000(X)
MBAPMAP :    80000116(X) ; VBIT,PAGE FRAME=000116(X)
RPCS1   :        0808(X) ; DVA,FUNCTION=DRIVE CLEAR
RPDS    :        0980(X) ; WRL,DPR,DRY
RPER1   :        0000(X) ;
RPER2   :        0000(X) ;
RPER3   :        0000(X) ;
RPCC    :        0000(X) ; CURRENT CYLINDER=00000(D)
RPOF    :        9000(X) ; SGCH,FMT22
RPDA    :        0002(X) ; TRACK=00(D),SECTOR=02(D)
RPLA    :        0540(X) ; SECTOR=21(D)
RPDC    :        0000(X) ; DESIRED CYLINDER=00000(D)
RPMR    :        1000(X) ; ZDI
RPDT    :        2011(X) ; MOH,DRIVE TYPE=PP05
RPAS    :        0000(X) ;
RPSN    :        0167(X) ; SERIAL NUMBER=0167(X)
RPEC1   :        0000(X) ; BURST LOCATION=0000(X)
RPEC2   :        0000(X) ; ERROR BURST=0000(X)

FINAL ERROR COUNT=   0
MAX   ERROR COUNT=   8

QIO STOP  TIME= 11:54:38.02
QIO START TIME= 11:54:38.02

  .. ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

Beispiel-Gerätefehlerausschrift (Kurzformat Ereignisflag 21=1):

```
******** PVRAD LEVEL 2 DISK FUNCTIONAL DIAGNOSTIC - 1.0 ********
 PASS 0 INITIALIZATION SECTION ERROR 9 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1:  INTERVENTION REQUIRED
```

```
RPDS            0980(X) ; WRL,DPR,DRY
RPER1           0000(X) ;
RPER2           0000(X) ;
RPER3           0000(X) ;
RPCC            0000(X) ; CURRENT CYLINDER=00000(D)
RPOF            9000(X) ; SGCH,FMT22
RPDA            0002(X) ; TRACK=00(D),SECTOR=02(D)

FINAL ERROR COUNT=   0
MAX   ERROR COUNT=   8

QIO STOP  TIME= 11:54:38.02
QIO START TIME= 11:54:38.02

 .. ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

6.3.5.7.2. Systemservicefehlerformat

Ein Systemservicefehler ist jeder Fehler, der entweder durch den Diagnose-Supervisor oder durch das Betriebssystem festgestellt wurde, während der FORMATTER eine Serviceroutine anfordert (z.B. ein $DS_GETBUF).

Beispiel einer Systemservicefehlermitteilung:

```
******** PVRAD LEVEL 2 DISK FUNCTIONAL DIAGNOSTIC - 1.0 ********
 PASS 0 INITIALIZATION SECTION ERROR 5 29-JAN-1987 11:54:38.02
 SYSTEM FATAL WHILE TESTING DBA1: FAILED TO OBTAIN BUFFER MEMORY

STATUS RETURN       PROCESS        DEFINITION
-----------------------------------------------------
00000244 VHS      VIRTUAL ADDRESS SPACE FULL

 .. ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

6.3.5.8. Programmitteilungen

Übersichtsausschriften werden am Ende jedes Durchlaufes für alle ausgewählten Geräte gegeben. Die Ausschrift enthält den Plattentyp, den Plattenkennungsnamen, den Schreibschutzstatus und den kumulativen Fehlerzähler des Gerätes.

6.3.5.9. Leistungsparameter des Testobjektes

entfällt

6.3.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.3.5.11. Testfolge bei mehreren Einheiten

entfällt

6.3.5.12. Einstellungen unter AFT

entfällt

6.3.5.13. Testsektionen

DEFAULT, NOWRITE

Das Funktionsdiagnoseprogramm deR Ebene 2 ist ein zerstörungsfreier Test. Das bedeutet, daß es die Daten, die vorher auf die Platte aufgezeichnet wurden, nicht zerstört. Da es die Notwendigkeit geben kann, schon dort zu testen, wo Anwender noch nutzen, darf diese Sektion abgearbeitet werden, die alle Schreibfunktionen unterdrückt.
Auch wenn das Testgerät nicht von jedem nutzbar ist, ist es schreibgeschützt. Dieser Status der Platte wird vom Programm erkannt. Es arbeitet dann entweder im Standard- oder im NOWRITE-Modus.

6.3.6. Testbeschreibungen

6.3.6.1. Test 1: PACK ACKNOWLEDGE TEST Gerätestatusabfrage

Dieser Test sendet ein PACK ACKNOWLEDGE(Stapelbestätigung-) Kommando. Nach der Kommandoprüfung auf korrekte Beendigung bringt das Gerät zur Bestätigung den gültigen Status: Datenträger gültig (VV-Bit), Gerät bereit, Gerät online usw.

Testschritte:

1. Beginn
2. PACK ACKNOWLEDGE
3. Wenn Fehler, dann Fehlerausschrift
4. Prüfung Gerätestatus
5. Wenn der geforderte Wert nicht kommt, dann Fehlerausschrift
6. Ende

6.3.6.2. Test 2: NOOP FUNCTION TEST Operationssimulation mit NOOP

Dieser Test erprobt, ob das Gerät die Fähigkeit hat, Funktionen über das Subsysteminterface auszuführen. Dies ist der Fall, wenn zehn NOOP-Funktionen

ausgeführt werden und bestätigt wird, daß der QIO mit Erfolg beendet wurde.

Testschritte:

1. Beginn
2. Wiederholen
3. NOOP-Funktion
4. Wenn Fehler, dann Fehlerausschrift
5. bis Schleifenzähler = 10
6. Ende

6.3.6.3. Test 3: DRIVE CLEAR FUNCTION TEST
Gerät rücksetzen

Senden eines "Gerät rücksetzen"-Kommandos und prüfen der entsprechenden Geräteregister, ob sie rückgesetzt sind.

Testschritte:

1. Beginn
2. wenn Gerät bereit, dann DRIVE CLEAR
3. Wenn Fehler, dann Fehlerausschrift
4. Prüfen, ob entsprechende Geräteregister rückgesetzt sind
5. Wenn Fehler, dann Fehlerausschrift
6. Ende

6.3.6.4. Test 4: READ IN PRESENT TEST
Lesevorbereitungstest

Dieser Test führt eine Lesevorbereitungs-Funktion aus und prüft, ob die betreffenden Geräteregister rückgesetzt sind.

Testschritte:

1. Beginn
2. Wenn das Gerät zu READ IN PRESENT bereit ist, dann READ IN PRESENT
3. Wenn Fehler, dann Fehlerausschrift
4. Prüfen, ob entsprechende Geräteregister rückgesetzt sind
5. Wenn Fehler, dann Fehlerausschrift
6. Ende

6.3.6.5. Test 5: RECAL FUNCTION TEST
Nullpositionieren

Dieser Test führt 10 Rücksetzfunktionen auf das Gerät aus. Da das Gerät bei der ersten RECAL-Operation auf Zylinder 0 positioniert wird, führt das Gerät eine minimale Bewegung aus, obwohl in diesem Fall der Anteil getesteter Logik beträchtlich höher ist.

Testschritte:

1. Beginn
2. Wenn das Gerät zur RECAL-Operation bereit ist, dann RECAL-Kommando
3. Wenn Fehler, dann Fehlerausschrift
4. bis Schleifenzähler = 10
5. Ende

6.3.6.6. Test 6: INCREMENTAL SEEK TEST Aufwärtspositionieren

Dieser Test positioniert das Gerät von Null an mit einem Zylindersuchkommando auf alle möglichen Zylinder

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Richtung = +1
5. Wiederholen
6. Positionieren auf aktuellen Zylinder + Richtung
7. Wenn Positionierfehler, dann Fehlerausschrift
8. Testen des speziellen Status
9. Wenn Fehler, dann Fehlerausschrift
10. Wenn Zylinder = Maximum, dann Richtung = -1
11. bis Zylinder = 0
12. Ende

6.3.6.7. Test 7: DIFFERENCE WORD TRANSMISSION TEST Alternierendes Suchen

Dieser Test positioniert das Gerät zwischen Zylinder 0 und allen Zylindradressen, die jedes binäre Adreßbit ergibt

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Wiederholen
5. Positionieren Gerät auf alle gültigen Binäradressen 2 hoch x (mit x=1, 2, 3 ...)
6. Wenn Positionierfehler, dann Fehlerausschrift
7. Prüfen des speziellen Gerätestatus
8. Wenn Fehler, dann Fehlerausschrift
9. bis Zylinder = maximal alle gültigen Binäradressen 2 hoch x (mit x=1, 2, 3 ...)
10. Ende

6.3.6.8. Test 8: SAWTOOTH SEEK TEST Sägezahnsuchen

Dieser Test prüft die Positionierleistung des Subsystems, indem alle möglichen Differenzpositionen vorwärts und rückwärts benutzt werden.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Wiederholen
5. Positionieren nach Zylinder + 1
6. Wenn Positionierfehler, dann Fehlerausschrift
7. Positionieren nach Zylinder 0
8. Wenn Positionierfehler, dann Fehlerausschrift
9. bis Zylinder = Maximalzylinder
10. Ende

6.3.6.9. Test 9: RANDOM SEEK TEST Zufälliges Positionieren

Dieser Test führt härteste Positionieroperationen aus, indem auf zufällige Adressen positioniert wird. Dies ist ein geeigneter Test, um flüchtige Positionierfehler zu provozieren, wenn er im MULTIPLE PASS MODE (zyklischer Test) benutzt wird.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Wiederholen
5. Positionieren auf zufällige Zylinderadresse
6. Wenn Positionierfehler, dann Fehlerausschrift
7. bis Schleifenzähler = 1024
8. Ende

6.3.6.10. Test 10: RANDOM RECAL TEST Zufälliges Nullpositionieren

Dieser Test prüft die Fähigkeit der Geräte zum Rücksetzen von verschiedenen zufälligen Zylindern.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Wiederholen
5. Positionieren zu zufälligen Zylinderadressen
6. Wenn Positionierfehler, dann Fehlerausschrift
7. RECAL Gerät
8. Wenn Rücksetzfehler, dann Fehlerausschrift

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Wiederholen
5. Suchen Sektor
6. Wenn Fehler, dann Fehlerausschrift
7. bis Sektor = Maximalsektor
8. Ende

6.3.6.12. Test 12: OFFSET-RETURN TO CENTERLINE TEST Offset- und Normalpositioniertest

Dieser Test prüft die Fähigkeit des Gerätes, von Normaloperationen mit dem OFFSET-Kommando in den OFFSET-Modus (Kopfversatz bei fehlerhaftem Lesen) überzugehen und mit dem RETURN TO CENTERLINE-Kommando zur Normaloperation zurückzukehren.

Testschritte:

1. Beginn
2. Wenn das Gerät nicht zur OFFSET-Funktion in der Lage ist, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
5. OFFSET-Kommando
6. Wenn OFFSET-Fehler, dann Fehlerausschrift
7. Prüfen des speziellen Gerätestatus
8. Wenn Fehler, dann Fehlerausschrift
9. RETURN TO CENTERLINE-Kommando
10. Wenn Fehler, dann Fehlerausschrift
11. Prüfen des speziellen Gerätestatus
12. Wenn Fehler, dann Fehlerausschrift
13. Ende

6.3.6.13. Test 13: FE CYLINDER ADDRESS TEST FE-Zylinderadreßtest

Dieser Test prüft die FE-Zylinderadressen von Festplatten.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Abbruch Test
4. Positionieren auf FE-Zylinder
5. Wenn Positionierfehler, dann Fehlerausschrift und Testabbruch
6. Prüfen der laufenden Zylinderregister
7. Wenn nicht korrekt, dann Fehlerausschrift und Testabbruch
8. Ende

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift
   und Abbruch Test
4. Positionieren auf FE-Zylinder
5. Wenn Positionierfehler, dann Fehlerausschrift und
   Testabbruch
6. Prüfen der laufenden Zylinderregister
7. Wenn nicht korrekt, dann Fehlerausschrift und Testabbruch
8. Ende

6.3.6.14. Test 14: BASIC DATA TEST Datenschreibgrundtest

Datengrundtest, bei dem die Fähigkeit des Gerätes, einen Sektor Daten korrekt zu schreiben und zu lesen, nachgewiesen wird.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Schreiben Daten
5. Wenn Fehler, dann Fehlerausschrift und Testabbruch
6. Lesen Daten
7. Wenn Fehler, dann Fehlerausschrift
8. Prüfen Daten
9. Wenn nicht korrekt, dann Datenvergleichsfehlerausschrift
10. Ende

6.3.6.15. Test 15: WRITECHECK ERROR TEST Provokation Schreibtestfehler

Test der Fähigkeit des Gerätes oder Subsystems (abhängig von der Konfiguration), Schreibtestfehler zu erkennen.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Wiederholen
5. Lesen Daten
6. Wenn Fehler, dann Fehlerausschrift und Testabbruch
7. Verfälschen der Daten im Hauptspeicher
8. Schreibtest Daten
9. Wenn kein Fehler, dann keine Schreibtestfehlerausschrift
10. bis alle 16 bit sich geändert haben
11. Ende

6.3.6.16. Test 16: WRITECHECK DATA TEST Datenschreibtestkommando

Test der Fähigkeit des Gerätes einen Sektor Daten korrekt mit Schreibtest zu prüfen

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Schreiben Daten mit erlaubtem Lesetest
5. Wenn Fehler, dann Lese/Schreibtestfehlerausschrift und Testabbruch
6. Ende

6.3.6.17. Test 17: HEADER AND DATA OPERATIONS TEST Kopf- und Datenoperation

Dieser Test prüft, ob das Gerät mit Kopfkommando lesen und Kopf mit Datenfunktionen miteinander kombinieren kann, wenn das Gerät so konzipiert ist.

Testschritte:

1. Beginn
2. Wenn das Gerät kein eingeschlossenes Positionieren kennt, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Lesen Kopf
6. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
7. Wenn Gerät nicht Kopf- und Datenoperationen kennt, dann Testabbruch
8. Wenn Fehler, dann Schreibtest- Kopf- und Datenfehlerausschrift
9. Ende

6.3.6.18. Test 18: SECTOR ADDRESSING TEST Sektoradressierung

Dieser Test prüft, ob das Gerät auf alle Sektoren einer gegebenen Spur zugreifen kann, unter Benutzung des Lese-Daten-Kommandos auf einzelne Sektoren in steigender Reihenfolge.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Sektor = 0
5. Wiederholen
6. Lesen Daten des laufenden Sektors
7. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
8. Prüfen Plattenadreßregister

9. Wenn Fehler, dann Fehlerausschrift
10. bis Sektor = Maximalsektor
11. Ende

6.3.6.19. Test 19: ZERO SECTOR FILL TEST Auffüllen eines nicht vollen Sektors mit Nullen

Test der Fähigkeit des Gerätes, bei gegebenem Schreibdatenkommando, mit kleinerem Bytezähler als ein Sektor groß ist, den Rest des Sektors mit Nullen aufzufüllen.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Schreiben Daten mit einem Bytezähler = 64 Worte
5. Wenn Fehler, dann Schreibfehlerausschrift und Testabbruch
6. Lesen Daten mit einem Bytezähler = voller Sektor
7. Wenn Fehler, dann Lesefehlerausschrift
8. Prüfen Daten
9. Wenn Fehler, dann Datenvergleichsfehlerausschrift
10. Ende

6.3.6.20. Test 20: MULTIPLE SECTOR TRANSFER TEST Lesen über Sektorengrenzen

Prüfen, ob das Subsystem Daten in mehreren Sektoren überträgt (über Sektorengrenzen), mit einem Sektorzähler = 2 und aufwärts laufend bis zur Summe der Sektoren -1.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Blockzähler = 2
5. Wiederholen
6. Lesen Daten mit erlaubter Datenprüfung
7. Wenn Fehler, dann Lese/Schreibtestfehlerausschrift und Testabbruch
8. bis Blockzähler = Zahl der Sektoren -1
9. Ende

6.3.6.21. Test 21: TRACK SELECTION TEST Spuradressierung

Prüfung, ob zu allen Spuren zugegriffen werden kann, ohne daß Kopf nicht gefunden oder andere Fehler auftreten.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Spur = 0
5. Wiederholen
6. Lesen Daten des laufenden Sektors
7. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
8. bis Spur = Maximalspur
9. Ende

6.3.6.22. Test 22: SPIRAL DATA TRANSFER TEST Lesen über Spurgrenzen

Dieser Test führt Blockübertragung in Gruppen von 2 Blöcken auf den letzten Sektor jeder Spur aus und prüft, ob das das Plattenadreßregister richtig hochgezählt hat (Lesen Daten über Spurgrenzen hinweg).

Testschritte:

1. Beginn
2. Wenn das Gerät nicht zum Datentranfer über Spurgrenzen hinweg in der Lage ist, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Spur = 0
6. Sektor = Maximalwert
7. Wiederholen
8. Lesen Daten zweier Sektoren
9. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
10. Wenn Plattenadreßregister falsch, dann Fehlerausschrift
11. bis Spur = Maximalspur -1
12. Ende

6.3.6.23. Test 23: IMPLIED SEEK TEST Lesen über Zylindergrenzen

Dieser Test prüft die Datenübertragung über die Zylindergrenzen hinweg. Es wird ein Zweiblock-Lesedatenkommando auf den letzten Sektor der letzten Spur eines zufälligen Zylinders gegeben.

Testschritte:

1. Beginn
2. Wenn das Gerät zu IMPLIED SEEK nicht in der Lage ist, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Spur = Maximalspur, Sektor = Maximalsektor
6. Lesen Daten des laufenden Sektors
7. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
8. Wenn Datenadreßregister oder Zylinderadresse falsch, dann Fehlerausschrift

9. Ende

6.3.6.24. Test 24: MAXIMUM DRIVE OPERATION TEST
Test der längsten Operation

Dieser Test prüft, ob das Gerät eine Funktion maximaler Zeitdauer ausführen kann. Dies wird beim Positionieren des Gerätes auf die maximale Spur des maximalen Zylinders erreicht. Dann wird ein eingeschlossenes Positionieren auf Zylinder 0 mit einem Bytezähler für eine volle Spur ausgeführt.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Positionieren Gerät auf Maximalspur des Maximalzylinders
5. Wenn Positionierfehler, dann Fehlerausschrift und Testabbruch
6. Lesen Spur auf Spur 0
7. Wenn Fehler, dann Lesefehlerausschrift
8. Ende

6.3.6.25. Test 25: LAST BLOCK TRANSFER TEST
Übertragung des letzten Blockes

Dieser Test prüft, ob eine Lese-Daten-Operation auf dem letzten Sektor das Setzen des Statusregisterbits "LAST BLOCK TRANSFERED (letzten Block übertragen)" bewirkt.

Testschritte:

1. Beginn
2. Wenn Subsystem kein LBT-Statusbit hat, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Lesen letzten Sektor
6. Wenn Fehler, dann Fehlerausschrift und Testabbruch
7. Prüfen des Gerätestatus
8. Wenn nicht korrekt, dann Fehlerausschrift
9. Ende

6.3.6.26. Test 26: ADDRESS OVERFLOW ERROR TEST
Adreßüberlauftest

Dieser Test prüft, ob eine Lese-Daten-Operation auf dem letzten Sektor mit einem Blockzähler von zwei Sektoren das Setzen des Statusregisterbits "ADDRESS OVERFLOW ERROR (Adreßüberlauffehler)" bewirkt.

Testschritte:

1. Beginn
2. Wenn Subsystem kein AOE-Statusbit hat, dann Testabbruch
3. Initialisieren Gerät

4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Lesen letzten Sektor mit Blockzähler = 2
6. Wenn kein Fehler, dann Fehlerausschrift und Testabbruch
7. Wenn kein AOE-Statusfehler, dann Fehlerausschrift
8. Ende

6.3.6.27. Test 27: SKIP SECTOR INHIBIT TEST
Sektorsprungtest

Test von Geräten, die Sektorensprungfähigkeit haben, auf Ausführung einer solchen Operation, wenn sie gefordert ist.

Testschritte:

1. Beginn
2. Wenn Gerät keine Sektorensprungfähigkeit hat, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Sektor = Maximalsektor
6. Lesen von 2 Blöcken mit Sektorsprung verboten
7. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
8. Prüfen des Gerätestatus auf "SKIP SECTORING WORKED"
9. Ende

6.3.6.28. Test 28: READ TRACK DESCRIPTOR TEST
Lesespur-Beschreiber-Test

Dieser Test führt auf zufällige Adressen Lesespur-Beschreiber-Operationen aus.

Testschritte:

1. Beginn
2. Wenn Gerät keine TDR-Funktionen ausführt, dann Testabbruch
3. Initialisieren Gerät
4. Wenn Fehler, dann Fehlerausschrift und Testabbruch
5. Wiederholen
6. Lesespur-Beschreiber
7. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
8. bis Zähler = 10
9. Ende

6.3.6.29. Test 29: RANDOM READ/WRITECHECK TEST
Lese-Schreibtest auf zufällige Adressen

Dieser Test führt Lese- und Schreibtest-Funktionen auf zufällige Adressen aus.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch

4. Wiederholen
5. Lesen Daten mit erlaubter Datenprüfung
6. Wenn Fehler, dann Lesefehlerausschrift und Testabbruch
7. bis Zähler = 512
8. Ende

6.3.6.30. Test 30: RANDOM FUNCTION TEST
Test zufälliger Kommandofolgen

Dieser Test versucht die Art zu simulieren, mit der SVP das Plattengerät mit zufälligen Kommandos anspricht. Die Kommandos sind Suchen, Schreiben, Schreibtest, Lesen und Positionieren. Geschrieben wird immer auf geschützte Bereiche.

Testschritte:

1. Beginn
2. Initialisieren Gerät
3. Wenn Fehler, dann Fehlerausschrift und Testabbruch
4. Wiederholen
5. Zufälliges Kommandoausführen
6. Wenn Fehler, dann Fehlerausschrift und Testabbruch
7. bis Schleifenzähler = 512
8. Ende

6.3.7. Auflistung der Tests

Test 01: PACK ACKNOWLEDGE TEST
Gerätestatusabfrage
Test 02: NOOP FUNCTION TEST
Operationssimulation mit NOOP
Test 03: DRIVE CLEAR FUNCTION TEST
Gerät rücksetzen
Test 04: READ IN PRESENT TEST
Lesevorbereitungstest
Test 05: RECAL FUNCTION TEST
Nullpositionieren
Test 06: INCREMENTAL SEEK TEST
Aufwärtspositionieren
Test 07: DIFFERENCE WORD TRANSMISSION TEST
Alternierendes Suchen
Test 08: SAWTOOTH SEEK TEST
Sägezahnsuchen
Test 09: RANDOM SEEK TEST
Zufälliges Positionieren
Test 10: RANDOM RECAL TEST
Zufälliges Nullpositionieren
Test 11: SEARCH FUNCTION TEST
Suchkommandotest
Test 12: OFFSET-RETURN TO CENTERLINE TEST
Offset- und Normalpositioniertest
Test 13: FE CYLINDER ADDRESS TEST
FE-Zylinderadreßtest
Test 14: BASIC DATA TEST
Datenschreibgrundtest

Test 15: WRITECHECK ERROR TEST
Provokation Schreibtestfehler
Test 16: WRITECHECK DATA TEST
Datenschreibtestkommando
Test 17: HEADER AND DATA OPERATIONS TEST
Kopf- und Datenoperation
Test 18: SECTOR ADDRESSING TEST
Sektoradressierung
Test 19: ZERO SECTOR FILL TEST
Auffüllen eines nicht vollen Sektors mit Nullen
Test 20: MULTIPLE SECTOR TRANSFER TEST
Lesen über Sektorengrenzen
Test 21: TRACK SELECTION TEST
Spuradressierung
Test 22: SPIRAL DATA TRANSFER TEST
Lesen über Spurgrenzen
Test 23: IMPLIED SEEK TEST
Lesen über Zylindergrenzen
Test 24: MAXIMUM DRIVE OPERATION TEST
Test der längsten Operation
Test 25: LAST BLOCK TRANSFER TEST
Übertragung des letzten Blockes
Test 26: ADDRESS OVERFLOW ERROR TEST
Adreßüberlauftest
Test 27: SKIP SECTOR INHIBIT TEST
Sektorsprungtest
Test 28: READ TRACK DESCRIPTOR TEST
Lesespur-Beschreiber-Test
Test 29: RANDOM READ/WRITECHECK TEST
Lese-Schreibtest auf zufällige Adressen
Test 30: RANDOM FUNCTION TEST
Test zufälliger Kommandofolgen

6.4. PVRBA: Kontrollerdiagnose mit Datenträger für PP05, PP06

6.4.1. Zusammenfassung

Das Programm PVRBA (Dok.-Nr. 1.57.700604.8) ist ein Diagnoseprogramm der Ebene 3 (Stand-alone-Modus). Das Programm besteht aus einer Testserie, welche die ordnungsgemäße Beschaffenheit der Wechselplattenspeicher CM 5404 bzw. CM 5416 nachweist. Das Programm enthält eine Reihe von Funktionstests mit folgendem Inhalt:

1) Datentransfer-Tests
2) Such-Tests (Positionierung)
3) Zeitablauf-Test
4) MANUAL-Tests

6.4.2. Testbedingungen

6.4.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- MSBUS-Adapter
- Wechselplattenspeicher PP05, PP06 (CM 5404, CM5416)

- Datenträger
  Das Programm fordert, daß die Platte aufliegt und sich das Gerät in der Stellung online befindet.

6.4.2.2. Software

- Diagnose-Supervisor über BOOT-Kommando gestartet.

6.4.3. Voraussetzungen

Der Test setzt voraus, daß alle PP0x Kontroller-Tests (PVRCA) fehlerfrei gelaufen sind und daß der MSBUS-Adapter arbeitsfähig ist. Es ist demzufolge erforderlich, daß die Kommandos, die sich nur auf Kontroller beziehen (HOUSEKEEPING), wie z.B. Gerät rücksetzen, Nullpositionieren, korrekt arbeiten. Diese Tests können nicht isoliert auf einer Modulebene versagen, infolge der Komplexität des analogen Lese/Schreib-Zyklusses.

6.4.4. Bedienungshinweise

Nachdem der Diagnose-Supervisor über BOOT-Kommando geladen und gestartet wurde, sind Kontroller und Gerät anzuschließen.

```
DS> ATTACH RHA40              ; Anschließen MSBUS-Adapter
DEVICE LINK? HUB              ;
DEVICE NAME? RHO              ;
```

```
TR? 8                          ;
BR? 5                          ;
DS> ATTACH PP05                ; Anschließen Wechselplattenspeicher
DEVICE LINK? RHO               ;
DEVICE NAME? DBA0              ;
DS> SELECT DBA0                ; Gerät auswählen
DS> LOAD PVRBA                 ;
DS> START                      ;
```

6.4.5. Funktionsbeschreibung

6.4.5.1. Übersicht

PVRBA läuft unter dem Diagnose-Supervisor und paßt sich den RVS-Diagnosebedingungen an. Es hat einen Kopfmodul, welcher Initialisierungs-, CLEAN-UP und eine Reihe anderer Subroutinen enthält. Außerdem sind hier alle globalen Text-, Daten- und Puffervereinbarungen enthalten.

Ein	Testmodul	enthält	die Datentransfertests	Test 1 ... 11
"	"	"	"	Test 12 ... 25
"	"	"	die Positioniertests	Test 26 ... 31
"	"	"	die Tests mit Handsteuerung	Test 32 ... 36
"	"	"	die Zeitablauftests	Test 37 ... 41

6.4.5.2. Programmgröße

Speicherbelegung von 200 (x) ... DDFF (x), das sind DC00 (x), d.h. 56320. byte.

6.4.5.3. Programmlaufzeit

Insgesamt ca. 10 min

6.4.5.4. Abarbeitungsbedingungen

Der Diagnose-Supervisor beginnt den Programmablauf immer mit der Initialisierungsroutine und schließt es mit der CLEAN-UP-Routine ab.

- Initialisierungsroutine

Die Routine ermittelt, wieviel Geräteeinheiten für den Test ausgewählt sind und versucht von jedem ausgewählten Gerät den Homeblock zu lesen. Bei Lesefehler oder wenn der Datenträgername nicht "SCRATCH" ist, meldet sich das Programm und fragt, ob der Test für dieses Gerät fortgesetzt werden soll. Für die Antwort "NO" wird diese Geräteeinheit nicht mehr getestet. Wird mit "YES" geantwortet, dann überschreibt das Diagnoseprogramm die gesamte Platte einschließlich des Homeblocks.

- CLEAN-UP-Code

In dieser Routine wird der Zustand überprüft, nachdem das Gerät getestet wurde. Bei einem Schreibversuch in den Homeblock mit dem Datenträgernamen "SCRATCH" wird nach einiger Zeit die Platte zu "SCRATCH" definiert. Dieser Testablauf wird für alle ausgewählten Geräte ausgeführt und ermöglicht damit die Schutzprüfung in der Initialisierungsroutine oder der Bediener übergeht die Schutzprüfung. Die Routine wird bis zum Ende eines Durchlaufs abgearbeitet oder der Bediener weist "ABORT" an bzw. der Makro "DS_ABORT" gelangt zur Ausführung.

6.4.5.5. Flagverwendung

entfällt

6.4.5.6. Allgemeine Programmanforderungen

Die manuellen Eingriffe sind in den jeweiligen Tests entsprechend den Operatoraufforderungen des Programmes vorzunehmen.

6.4.5.7. Fehlermitteilung

- Statusfehler

```
Test 3: WRITE HEADER & DATA, READ HEADER & DATA TESTS
***** PVRBA PP0X FUNCTIONAL DIAGNOSTIC - 1.0 *****
Pass 1, test 3, subtest 2, error 3, 12-MAI-1987 09:28:59.12
Hard error while testing DBA0: STATUS ERROR (ERR BIT SET)

REGISTER                  DATA              MNEMONICS

RPCS1 [40F28400]:         083A(X)           ; DVA, FUNKTION=READ, HEADER & DATA
RPDS  [40F28404]:         D1C0(X)           ; ATA, ERR,MOL,DPR,DRY,VV
RPER1 [40F28408]:         00B0(X)           ; HCE
RPMR  [40F2840C]:         0100(X)           ; ZDT
RPAS  [40F28410]:         0001(X)           ; ATA0
RPDA  [40F28414]:         0001(X)           ; TRACK=00(D),SEKTOR=01(D)
RPDT  [40F28418]:         2012(X)           ; MOH,DRIVE TYPE=PP06
RPLA  [40F2841C]:         04A0(X)           ; SECTOR=18(D), EXT1
RPSN  [40F28420]:         0000(X)           ;
RPOF  [40F28424]:         9000(X)           ; SGCH, EMT22
RPDC  [40F28428]:         0000(X)           ; DES/RED CYLINDER=0000(D)
RPCC  [40F2842C]:         0000(X)           ; CURRENT CYLINDER=0000(D)
RPER2 [40F28430]:         0000(X)           ;
RPER3 [40F28434]:         0000(X)           ;
RPEC1 [40F28438]:         0000(X)           ;
RPEC2 [40F2843C]:         0000(X)           ;

MBA CHANNEL STATUS DUMP

MBA_CSR: [00F28000]       00000000(X)       ;
MBA_CR:  [00F28004]       00000000(X)       ;
MBA_SR:  [00F28008]       00013400(X)       ; ATTN, DTCOMP, DTABT, MDPE
MBA_VAR: [00F2800C]       00000002(X)       ;
```

MBA_MAP(40): 00000000(X)
MBA_BCNT:[00F28010] FDFCFDFA(X)

***** End of Hard error number 3 *****

6.4.5.8. Programmitteilungen

entfällt

6.4.5.9. Leistungsparameter des Testobjektes

entfällt

6.4.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.4.5.11. Testfolge bei mehreren Einheiten

Bei mehreren Einheiten werden sämtliche ausgewählten Tests zunächst für eine Einheit abgearbeitet. Im Anschluß daran die gleiche Testfolge für die nächste Einheit. Die Auswahl der Einheiten erfolgt in der Reihenfolge der Selektierung.

6.4.5.12. Einstellungen für den Ablauf unter AFT

entfällt

6.4.5.13. Testsektionen

PVRBA PP0X FUNCTIONAL DIAGNOSTIC enthält folgende Sektionen:
DEFAULT, DATA, MANUAL, TIMING, POSITION, SEEK_TST, ALL, ACCESS

1) DEFAULT - Standardtestsektion
   Hier sind alle Tests, außer solchen mit manuellen Eingriffen enthalten, also Test 1-31 und 38-41

2) DATA - Datentransfertestsektion
   Test 2-25

3) MANUAL - Testsektion mit manuellen Eingriffen
   Test 32-36

4) TIMING - Zeitablauftestsektion
   Test 38-41

5) POSITION - Positioniertestsektion
Test 26-31

6) SEEK - Positioniertestsektion
Test 26-31

7) ALL - Testsektion mit sämtlichen Tests
Test 1-41

8) ACCESS - reserviert

6.4.5.14. Verschiedenes

entfällt

6.4.6. Testbeschreibungen

6.4.6.1. Test 1: NOOP COMMAND Keine Operation

Der Test überprüft die ordnungsgemäße Abarbeitung des Kommandos "keine Operation". Das Kommando wird ausgeführt und danach alle Register überprüft.

Testschritte:

1. MSBUS initialisieren
2. Kommando keine Operation anweisen
3. Vergleich, ob sich nichts verändert hat

Testmethodik:

Der Test überprüft, ob sich die Register nach Ausführung des Kommandos verändert haben. Wenn der Registerinhalt nach diesem Kommando wechselt, dann wird die Funktion wahrscheinlich nicht ordnungsgemäß entschlüsselt. Prüfen der Logik des Funktionsdekoders der RG-BLP. Eine andere Möglichkeit ist, daß die Registerauswahllogik nicht richtig arbeitet. Die Registerauswahllogik wird beeinflußt von der IC-BLP und der RG-BLP.

Fehler:

Fehler 1: Kontroll- und Statusregister halten 1 bit fest
Fehler 2: Gerätestatus falsch (DRY,DPR,MOL gesetzt, ERR gelöscht)

6.4.6.2. Test 2: DRIVE CLEAR Gerät rücksetzen

Dieser Test schreibt alles 1 in ein jedes R/W-Register und führt dann das "Gerät rücksetzen" aus. Es wird verglichen, ob alle Bits gelöscht sind.

Testschritte:

1. Vergleich, ob MOL-Bit gesetzt (DRIVER-CLEAR wird nicht ausgeführt, wenn das Bit gesetzt ist)
2. Die Register RPER1, RPOF, RPER2, RPER3 werden mit 1 geladen
3. Kommando "Gerät rücksetzen" ausführen
3. Vergleich, ob obige Register gelöscht sind

Testmethodik:

Das Kommando "Gerät rücksetzen" wird von der RG-BLP dekodiert. Wenn der MBA INIT-Test keinen Fehler hatte, dann wird das Kommando nicht richtig dekodiert. Falls ein Bit in den Registern RPER2 oder RPER3 festgehalten wird, kann eine harte Fehlerbedingung im Gerät vorliegen.

6.4.6.3. Test 3: WRITE HEADER & DATA,
READ HEADER & DATA TESTS
Schreiben Kopf & Daten,
Lesen Kopf & Daten

Subtest 1:

Es werden Kopf und Daten nach Zylinder 0, Spur 0, Sektor 0 mit 512 Datenbytes (alles 0) geschrieben (Bytezählerlänge=520). Der Subtest 1 setzt voraus, daß das Gerät online und das Bit VOL VALID gesetzt ist.

Subtest 2:

Es werden Kopf und Daten von Zylinder 0, Spur 0, Sektor 0 mit der Bytezählerlänge 520 gelesen. Der Subtest 2 wird nicht durchgeführt, wenn bei Subtest 1 Fehler auftraten (WRITE HEADER & DATA - Kommando fehlerhaft). Der Subtest 2 setzt voraus, daß das Gerät online und der aufgelegte Datenträger verfügbar ist und das Schreiben in Subtest 1 ordnungsgemäß ausgeführt wurde.

Testschritte:

Subtest 1:
1. Prüfen des Bit VOL VALID, MOL usw.
2. Einstellen der Plattenadresse
3. Einstellen des Map-Registers
4. Schreiben von Kopf & Daten anweisen
5. Warten auf Ende der Kommandoausführung
6. Prüfen auf Statusfehler
7. Prüfen, ob Gerät Timeout-Fehler bringt

Subtest 2:
1. Einstellen der Plattenadresse
2. Einstellen des Map-Registers
3. Lesen Kopf & Daten anweisen
4. Prüfen auf Statusfehler
5. Wenn kein Fehler auftritt dann Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: VOL VALID-Bit rückgesetzt

Fehler 3: Statusfehler nach Schreiben
Fehler 4: Statusfehler nach Lesen
Fehler 5: Datenvergleichsfehler

Testmethodik:

Das Testergebnis ist von der HEADER-Vergleichslogik unabhängig und befindet sich im sicheren Bereich der Positionierlogik.
Wenn bei dem Test Fehler auftreten, ist es schwierig auszusagen, ob es sich um einen Schreib- oder Lesefehler handelt. Es ist in diesem Fall besser abzubrechen und das Diagnoseprogramm PVRCA zu starten, um die Plattenkontroller-Logik zu überprüfen. Bei Nutzung von PVRCA ist es leichter, einen Schreib- oder Lesefehler im Plattenkontroller zu isolieren.

6.4.6.4. Test 4: WRITE/READ DATA TESTS
Daten Schreiben/Lesen

Der Test 4 erfordert, daß das Gerät online und der Datenträger verfügbar ist.

Subtest 1:

Daten nach Zylinder 0, Spur 0, Sektor 0 mit 512 Datenbyte alles Null schreiben.

Subtest 2:

512 byte von Zylinder 0, Spur 0, Sektor 0 lesen und die Daten mit denen im Schreibpuffer des vorangegangenen Subtests vergleichen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen der Plattenadresse
3. Einstellen des Map-Registers
4. Daten schreiben anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Plattenadresse
3. Einstellen Map-Register
4. Daten lesen anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Wenn kein Fehler Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: VOL VALID-Bit rückgesetzt
Fehler 3: Statusfehler nach Schreiben
Fehler 4: Statusfehler nach Lesen
Fehler 5: Datenvergleichsfehler

Testmethodik:

Der Test 4 ist ähnlich dem vorangegangenen Test 3, ausgenommen, daß die HEADER-Vergleichslogik verwendet werden muß. Der verwendete Sektor wurde im vorangegangenen Test formatiert. Wenn ein HEADER-Fehler (HCE, HCRC, FMT) auftritt, so wird dies durch ein Schreibproblem hervorgerufen, welches sich im vorangegangenen Test ereignet hat.

6.4.6.5. Test 5: WRITE/READ DATA TESTS Daten Schreiben/Lesen

Der Test 5 erfordert, daß das Gerät online und der Datenträger verfügbar ist.

Subtest 1:

Daten nach Zylinder 0, Spur 0, Sektor 0 mit 400 Datenbyte alles 1 und 112 byte AA(x) schreiben.

Subtest 2:

512 byte von Zylinder 0, Spur 0, Sektor 0 lesen und die Daten mit denen im Schreibpuffer des vorangegangenen Subtests vergleichen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen der Plattenadresse
3. Einstellen des Map-Registers
4. Daten schreiben anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Plattenadresse
3. Einstellen Map-Register
4. Daten lesen anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Wenn kein Fehler Datenvergleich

Fehler:

Fehler 2: VOL VALID-Bit rückgesetzt
Fehler 3: Statusfehler nach Schreiben
Fehler 4: Statusfehler nach Lesen
Fehler 5: Datenvergleichsfehler

Testmethodik:

Der Test 5 ist ähnlich dem vorangegangenen Test 3, ausgenommen, daß die HEADER-Vergleichslogik verwendet werden muß. Der verwendete Sektor wurde im vorangegangenen Test formatiert. Wenn ein HEADER-Fehler (HCE, HCRC, FMT) auftritt, so wird dies durch ein Schreibproblem hervorgerufen, welches sich im vorangegangenen Test ereignet hat.

6.4.6.6. Test 6: WRITE/READ DATA TESTS Daten Schreiben/Lesen

Der Test 6 erfordert, daß das Gerät online und der Datenträger verfügbar ist.

Subtest 1:

Daten nach Zylinder 0, Spur 0, Sektor 0 mit 512 Datenbyte alles AA(x) schreiben.

Subtest 2:

512 byte von Zylinder 0, Spur 0, Sektor 0 lesen und die Daten mit denen im Schreibpuffer des vorangegangenen Subtests vergleichen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen der Plattenadresse
3. Einstellen des Map-Registers
4. Daten schreiben anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Plattenadresse
3. Einstellen Map-Register
4. Daten lesen anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Wenn kein Fehler Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: VOL VALID-Bit rückgesetzt
Fehler 3: Statusfehler nach Schreiben
Fehler 4: Statusfehler nach Lesen
Fehler 5: Datenvergleichsfehler

Testmethodik:

Der Test 6 ist ähnlich dem vorangegangenen Test 3, ausgenommen, daß die HEADER-Vergleichslogik verwendet werden muß. Der verwendete Sektor wurde im vorangegangenen Test formatiert. Wenn ein HEADER-Fehler (HCE, HCRC, FMT) auftritt, so wird dies durch ein Schreibproblem hervorgerufen, welches sich im vorangegangenen Test ereignet hat.

6.4.6.7. Test 7: WRITE/READ DATA TESTS Daten Schreiben/Lesen

Der Test 7 erfordert, daß das Gerät online und der Datenträger verfügbar ist.

Subtest 1:

Daten nach Zylinder 0, Spur 0, Sektor 0 mit 512 Datenbyte alles 55(x) schreiben.

Subtest 2:

512 byte von Zylinder 0, Spur 0, Sektor 0 lesen und die Daten mit denen im Schreibpuffer des vorangegangenen Subtests vergleichen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen der Plattenadresse
3. Einstellen des Map-Registers
4. Daten schreiben anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Plattenadresse
3. Einstellen Map-Register
4. Daten lesen anweisen
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Wenn kein Fehler Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: VOL VALID-Bit rückgesetzt
Fehler 3: Statusfehler nach Schreiben
Fehler 4: Statusfehler nach Lesen
Fehler 5: Datenvergleichsfehler

Testmethodik:

Der Test 7 ist ähnlich dem vorangegangenen Test 3, ausgenommen, daß die HEADER-Vergleichslogik verwendet werden muß. Der verwendete Sektor wurde im vorangegangenen Test formatiert. Wenn ein HEADER-Fehler (HCE, HCRC, FMT) auftritt, so wird dies durch ein Schreibproblem hervorgerufen, welches sich im vorangegangenen Test ereignet hat.

6.4.6.8. Test 8: IMPLIED SEARCH TEST
Datentransfer einschließlich Suchen

Der Test veranlaßt einen Datentransfer einschließlich Suchen, da durch den Datentransfer die Sektorgrenzen überschritten werden. Kopf und Daten werden nach Zylinder 0, Spur 0, Sektor 0 mit 532 byte geschrieben. Diese Bytes haben folgende Bedeutung:

8 byte HEADER, 512 Datenbyte alle 0, 8 byte HEADER, 4 byte alle 1

Dann werden Kopf und Daten von der gleichen Adresse gelesen. Falls kein Fehler vorliegt, werden anschließend die Daten verglichen.

Testschritte:

Subtest 1:
1. Einstellen der Plattenadresse
2. Initialisieren
3. Einstellen des Map-Registers
4. Einstellen Plattenkontroller-Register
5. Schreiben von Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung

Subtest 2:
1. Einstellen der Plattenadresse
2. Initialisieren
3. Einstellen des Map-Registers
4. Einstellen Plattenkontroller-Register
5. Lesen von Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung
8. Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: Statusfehler nach Schreiben
Fehler 3: Statusfehler nach Lesen
Fehler 4: Datenvergleichsfehler

Testmethodik:

Das ist der erste Test, der versucht, auf mehr als einen Sektor zuzugreifen. Falls der Test Fehler bringt, die vorangegangenen Tests aber nicht, dann liegt das Problem wahrscheinlich im Bereich der Zeitablaufsteuerung des Plattenspeichers. Eventuell wurde die Zeitablaufsteuerung nicht zeitig genug rückgesetzt. Es wird also verglichen, daß die Sektorimpulse ordnungsgemäß dekodiert (von der DP-BLP) werden (besonders bei der Fehlermeldung "OPI").

6.4.6.9. Test 9: IMPLIED SEEK TEST Datentransfer einschließlich Positionieren

Der Test veranlaßt einen Datentransfer einschließlich Positionieren, da durch den Datentransfer die Sektorgrenzen überschritten werden. Kopf und Daten werden nach Zylinder 0, Spur 18, Sektor 21 mit 532 byte geschrieben. Diese Bytes haben folgende Bedeutung:

8 byte HEADER, 512 Datenbyte alle 0, 8 byte HEADER, 4 byte alle 1

Dann werden Kopf und Daten von der gleichen Adresse gelesen. Falls kein Fehler vorliegt, werden anschließend die Daten verglichen.

Testschritte:

Subtest 1:
1. Einstellen der Plattenadresse
2. Initialisieren

3. Einstellen des Map-Registers
4. Einstellen Plattenkontroller-Register
5. Schreiben von Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung

Subtest 2:
1. Einstellen der Plattenadresse
2. Initialisieren
3. Einstellen des Map-Registers
4. Einstellen Plattenkontroller-Register
5. Lesen von Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung
8. Datenvergleich

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: Statusfehler nach Schreiben
Fehler 3: Statusfehler nach Lesen
Fehler 4: Datenvergleichsfehler

Testmethodik:

Dies ist der erste Test, der ein MID-Transfer-Suchen (Datentransfer über Zylindergrenze hinaus) versucht. Bringt der Test Fehler, funktioniert wahrscheinlich das Servosystem nicht richtig, besonders wenn "SKI"-Fehler vorliegt. Man vergleiche das mit der Arbeit der Positionier-Tests.

6.4.6.10. Test 10: SEEK AND WRITE TEST
Positionieren und Schreiben

Dieser Test positioniert die Köpfe durch 10maliges einschließliches Positionieren über Zylinder 1 bis Zylinder 10. Der Test beginnt mit einem RECALIBRATE-Kommando, um die Köpfe zurück auf Zylinder 0 zu positionieren. Dann werden Kopf und Daten ab Zylinder 0, Spur 18, Sektor 21 mit 532 byte Länge geschrieben, wodurch die Köpfe nach Zylinder 1 positioniert werden. Dieser Ablauf wird für die restlichen Zylinder wiederholt, so daß die Köpfe schließlich zu Zylinder 10 positioniert werden. Dann werden die Daten zurückgelesen und verglichen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Rücksetzen auf Zylinder 0
3. Schreibpuffer füllen
4. Einstellen des Map-Registers
5. Einstellen Plattenadresse
6. Einstellen Plattenkontroller-Register
7. Starten des Datentransfer "Schreiben"
8. Warten auf Ende der Kommandoausführung
9. Prüfen auf Statusfehler
10. Wiederholen bis Zylinder 10 erreicht ist

Subtest 2:
1. Initialisieren
2. Zurück nach Zylinder 0
3. Einstellen Map-Register
4. Einstellen Plattenadresse
5. Einstellen PLattenkontroller-Register
6. Starten des Datentransfer "Lesen"
7. Warten auf Ende der Kommandoausführung
8. Prüfen auf Statusfehler
9. Wenn kein Fehler, dann Datenvergleich
10. Wiederholen bis Zylinder 10 erreicht ist

Fehler:

Fehler 1: DRY, MOL, VV nicht gesetzt
Fehler 2: RECALIBRATE-Funktion fehlerhaft
Fehler 3: Gerät Zeitfehler
Fehler 4: Statusfehler

Testmethodik:

entfällt;

6.4.6.11. Test 11: INTERRUPT TEST Interrupterkennung

Der Test erwartet in den Subtests 1 bis 3 Interrupte nach Ausführung der Kommandos WRITE, READ, RECALIBRATE (ATTENTION Signal ein).

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenadresse
4. Einstellen Plattenkontroller-Register
5. Interrupt erlauben
6. Schreib-Kommando geben
7. Warten auf Interrupt
8. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenadresse
4. Einstellen Plattenkontroller-Register
5. Interrupt erlauben
6. Lese-Kommando geben
7. Warten auf Interrupt
8. Fehlerprüfung

Subtest 3:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenadresse
4. Einstellen Plattenkontroller-Register

5. Interrupt erlauben
6. Nullposition-Kommando geben
7. Warten auf Interrupt
8. Fehlerprüfung

Fehler:

Fehler 1: kein Interrupt
Fehler 2: Statusfehler

Testmethodik:

Es ist nicht notwendig im Plattenkontroller nachzusehen. Die gesamte Interruptlogik befindet sich im MSBUS-Adapter.

6.4.6.12. Test 12: WRITE CHECK HEADER & DATA TEST Schreiben Kopf und Daten mit Prüfung

Der Test führt ein Schreiben von Kopf und Daten mit Prüfung nach Zylinder 5, Spur 7, Sektor 4 mit 532 byte Länge aus. Mit folgenden Datenmustern wird der Schreibpuffer aufbereitet:

8	byte	HEADER
10x2	byte	14E4
20	byte	alles 1
20	byte	alles 0
20	byte	55
20	byte	AA
32	byte	durchlaufende 0
32	byte	durchlaufende 1
174x2	byte	FF00
8	byte	HEADER
4	byte	14E5

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Schreib-Puffer
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Schreiben Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Lesen Kopf und Daten
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Datenvergleich

Subtest 3:
Bei Fehler der vorangegangenen Subtests erfolgt Abbruch
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Schreiben einschließlich Prüfung Kopf und Daten
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Testmethodik:

Die Logik der Schreibprüfung befindet sich immer im MSBUS-Adapter RHA40. Ist der Test fehlerhaft, aber die Schreib/Lese-Tests arbeiten richtig, dann liegt der Fehler wahrscheinlich auch im MSBUS-Adapter.

6.4.6.13. Test 13: WRITE CHECK DATA TEST
Daten schreiben mit Prüfung

Der Test führt ein Datenschreiben mit Prüfung nach Zylinder 5, Spur 7, Sektor 4 mit 516 byte Länge durch. Mit folgenden Datenmustern wird der Schreibpuffer aufbereitet:

10*2	byte alles 14E4
20	byte alles 1
20	byte alles 0
20	byte 55
20	byte AA
32	byte durchlaufende 0
32	byte durchlaufende 1
174*2	byte FF00
4	byte 14E5

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Schreib-Puffer
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Schreiben Kopf und Daten
6. Warten auf Ende der Kommandoausführung
7. Fehlerprüfung

Subtest 2:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Lesen Kopf und Daten
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung
7. Datenvergleich

Subtest 3:
Bei Fehler der vorangegangenen Subtests erfolgt Abbruch
1. Initialisieren
2. Einstellen Map-Register

3. Einstellen Plattenkontroller-Register
4. Schreiben einschließlich Prüfung Kopf und Daten
5. Warten auf Ende der Kommandoausführung
6. Fehlerprüfung

Testmethodik:

Der Test ist dem vorangegangenen sehr ähnlich, ausgenommen die Kommandos Daten schreiben und lesen sowie Daten schreiben mit Prüfung. Wenn der vorherige Test richtig lief, dieser Test aber fehlerhaft, dann wurden wahrscheinlich die Kommandos nicht richtig entschüsselt.

6.4.6.14. Test 14: LAST SECTOR TRANSFERRED "LST" TEST Datentransfer mit dem letzten Sektor

Es werden 512 byte nach Zylinder 814, Spur 18, Sektor 21 geschrieben. Dann wird verglichen, ob das LST-Bit gesetzt ist. Nach Rücksetzen des Gerätes erfolgt die gleiche Prozedur mit dem Lesekommando.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Datenpuffer
3. Einstellen Map-Register
4. Einstellen des Plattenkontroller-Registers
5. Ausführung des Datenschreib-Kommandos
6. Warten auf das Ende der Kommandoausführung
7. Prüfen LST-Bit
8. Prüfen auf andere Fehler

Subtest 2:
1. Initialisieren
2. Einstellen Datenpuffer
3. Einstellen Map-Register
4. Einstellen des Plattenkontroller-Registers
5. Ausführung des Datenlese-Kommandos
6. Warten auf das Ende der Kommandoausführung
7. Prüfen LST-Bit
8. Prüfen auf andere Fehler

Fehler:

Fehler 1: LST-Bit nicht gesetzt
Fehler 2: Gerätezeitfehler
Fehler 3: Andere Statusfehler

Testmethodik:

entfällt;

6.4.6.15. Test 15: ADRESS OVERFLOW ERROR "AOE" TEST Fehlerprovokation Adreß-überlauf

Der Test 15 führt ein Schreiben von 513 byte nach Zylinder 814, Spur 18, Sektor 21 aus. Danach wird verglichen, ob das AOE-Bit und das LST-Bit gesetzt sind. Nach Rücksetzen des Gerätes erfolgt die gleiche Prozedur für das Lesen.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Datenpuffer
3. Einstellen Map-Register
4. Einstellen des Plattenkontroller-Registers
5. Ausführung des Datenschreib-Kommandos
6. Warten auf das Ende der Kommandoausführung
7. Prüfen AOE-Bit und LST-Bit
8. Prüfen auf andere Fehler

Subtest 2:
1. Initialisieren
2. Einstellen Datenpuffer
3. Einstellen Map-Register
4. Einstellen des Plattenkontroller-Registers
5. Ausführung des Datenlese-Kommandos
6. Warten auf das Ende der Kommandoausführung
7. Prüfen AOE-Bit und LST-Bit
8. Prüfen auf andere Fehler

Fehler:

Fehler 1: AOE-Bit und LST-Bit nicht gesetzt
Fehler 2: Gerätezeitfehler
Fehler 3: Andere Statusfehler

Testmethodik:

entfällt;

6.4.6.16. Test 16: INVALID ADDRESS ERROR TEST Fehlerprovokation falsche Adresse

Subtest 1: Zylinder zu groß

Durch ein Schreiben von Kopf und Daten wird versucht auf Zylinder 411 (PP05) bzw. 815 (PP06), Spur 0, Sektor 0 zuzugreifen. Das IAE-Bit muß dann gesetzt sein.

Subtest 2: Spur zu groß

Mit einem Datenschreib-Kommando wird versucht auf Zylinder 0, Spur 19, Sektor 0 zuzugreifen. Es muß wiederum das IAE-Bit gesetzt sein.

Subtest 3: Sektor zu groß

Ein Kopf & Datenlese-Kommando versucht auf Zylinder 0, Spur 0, Sektor 22 zuzugreifen. Das IAE-Bit muß ebenfalls gesetzt werden.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Vergleich, ob IAE-Bit gesetzt
5. Vergleich, ob EXC-Bit gezetzt
6. Vergleich, ob andere Statusfehler vorliegen

Subtest 2:
wie Subtest 1 für Datenschreib-Kommando

Subtest 3:
wie Subtest 1 für Kopf & Datenlese-Kommando

Fehler:

Fehler 1: Kein IAE-Bit gesetzt
Fehler 2: Kein EXC-Bit gesetzt
Fehler 3: Andere Statusfehler
Fehler 4: Gerät nicht bereit

Testmethodik:

Es wird die illegale Adressdekodierung überprüft. Diese befindet sich auf der SS-BLP.

6.4.6.17. Test 17: FORMAT ERROR TEST Fehlerprovokation falsches Format

Der Test 17 ist ein begrenzter Test für das FORMAT ERROR-Bit, weil es den 18-bit-Mode nicht unterstützt. Es wird versucht, nach Zylinder 0, Spur 0, Sektor 0 im 18-bit-Mode (FMT22-Bit rückgesetzt) zu schreiben, obwohl die Kopfinformation für 16 bit formatiert ist. Es wird verglichen, ob das FER-Bit gesetzt und im MBA das EXCEPTION-Bit gesetzt ist. Die gleiche Prozedur wird mit dem Datenlese-Kommando wiederholt.

Testschritte:

Subtest 1:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Löschen FMT22-Bit
5. Setzen GO Bit (Datenschreib-Kommando ausführen)
6. Status prüfen
7. Vergleich, ob FER-Bit gesetzt
8. Vergleich, ob EXC-Bit gesetzt

Subtest 2:
1. Initialisieren
2. Einstellen Map-Register
3. Einstellen Plattenkontroller-Register
4. Löschen FMT22-Bit
5. Setzen GO-Bit (Datenlese-Kommando ausführen)
6. Status prüfen
7. Vergleich, ob FER-Bit gesetzt
8. Vergleich, ob EXC-Bit gesetzt

Fehler:

Fehler 1: FER-Bit nicht gesetzt
Fehler 2: EXC-Bit nicht gesetzt
Fehler 3: Andere Statusfehler

Testmethodik:

entfällt;

6.4.6.18. Test 18: ILLEGAL FUNCTION TEST
Fehlerprovokation illegale
Funktion

Dieser Test vergleicht, ob alle illegalen Funktionskodes den Fehler "Illegale Funktion" erzeugen. Nachfolgend eine Liste der illegalen Funktionskodes:

0A, 0B, 0D, 0F, 10, 11, 12, 13, 16, 17, 1A, 1B, 1E, 1F

Testschritte:

1. Initialisieren
2. Funktionskode nach CS1-Register laden
3. Setzen GO-Bit (Funktionsausführung)
4. Prüfen ILF-Bit
5. Wiederholen für alle illegalen Funktionskodes

Fehler:

Fehler 1: ILF-Bit bleibt 0

Testmethodik:

Prüfen der Funktionentschlüsselung die RG-BLP. Es wird also verglichen, ob das zuständige Flip-Flop einschaltet.

6.4.6.19. Test 19: HEADER COMPARE ERROR
Fehlerprovokation
falsche Kopfinformation

Zylinder 0, Spur 0 Sektor 1 wird als Zylinder 0, Spur 0, Sektor 0 formatiert. Ein Datenlese-Kommando versucht auf Sektor 1 zuzugreifen. Es muß danach das HCE-Bit gesetzt sein. Analog wird dies mit dem Datenschreib-Kommando durchgeführt.

Zum Abschluß wird Sektor 1 wieder ordnungsgemäß formatiert. Um sicher zu sein, daß Sektor 1 richtig formatiert wird, erfolgt dies in der CLEAN-UP-Routine.

Testschritte:

Subtest 1:
1. Sektor 1 fehlerhaft formatieren
2. Initialisieren
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Datenlese-Kommando durchführen
6. Prüfen, ob HCE-Bit gesetzt
7. Prüfen, ob MBA EXCEPTION-Bit gesetzt

Subtest 2:
1. Sektor 1 fehlerhaft formatieren
2. Initialisieren
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Datenschreib-Kommando durchführen
6. Prüfen, ob HCE-Bit gesetzt
7. Prüfen, ob MBA EXCEPTION-Bit gesetzt
8. Sektor 1 Format reorganisieren

Fehler:

Fehler 1: HCE-Bit nicht gesetzt
Fehler 2: MBA EXCEPTION-Bit nicht gesetzt
Fehler 3: Andere Statusfehler

Testmethodik:

Die HEADER-Vergleichslogik befindet sich auf der SS-BLP.

6.4.6.20. Test 20: HEAD SELECTION TEST Spurauswahl aus der Kopfinformation

Der Test überprüft die HEAD SELECTION-Logik. Ein Kopf & Datenschreib-Kommando wird für Zylinder 0, Spur 0-18, Sektor 0 ausgeführt. Danach folgt für die gleichen Positionen ein Kopf & Datenlese-Kommando mit anschließendem Datenvergleich.

Testschritte:

1. Initialisieren
2. Schreibpuffer füllen
3. Einstellen Map-Register für Schreiben
4. Einstellen Plattenkontroller-Register für Schreiben
5. Kopf & Datenschreib-Kommando ausführen
6. Einstellen Map-Register für Lesen
7. Einstellen Plattenkontroller-Register für Lesen
8. Kopf & Datenlese-Kommando ausführen
9. Datenvergleich
10. Wiederholen für alle Spuren 0-1

Fehler:

Fehler 1: Statusfehler Schreiben
Fehler 2: Statusfehler Lesen
Fehler 3: Gerät nicht bereit
Fehler 4: Datenvergleichsfehler

Testmethodik:

Die HEAD SELECTION-Logik befindet sich auf der SS-BLP.

6.4.6.21. Test 21: SECTOR SELECTION TEST Sektorauswahl aus der Kopfinformation

Der Test überprüft die HEAD SELECTION-Logik. Ein Kopf & Datenschreib-Kommando wird für Zylinder 0, Spur 0, Sektor 0-21 ausgeführt. Danach folgt für die gleichen Positionen ein Kopf & Datenlese-Kommando mit anschließendem Datenvergleich.

Testschritte:

1. Initialisieren
2. Schreibpuffer füllen
3. Einstellen Map-Register für Schreiben
4. Einstellen Plattenkontroller-Register für Schreiben
5. Kopf und Datenschreib-Kommando ausführen
6. Einstellen Map-Register für Lesen
7. Einstellen Plattenkontroller-Register für Lesen
8. Kopf und Datenlese-Kommando ausführen
9. Datenvergleich
10. Wiederholen für alle Sektoren (0-21)

Fehler:

Fehler 1: Statusfehler Schreiben
Fehler 2: Statusfehler Lesen
Fehler 3: Gerät nicht bereit
Fehler 4: Datenvergleichsfehler

Testmethodik:

Die HEAD SELECTION-Logik befindet sich auf der SS-BLP.

Testschritte:

1. Initialisieren
2. Schreibpuffer füllen
3. Einstellen Map-Register für Schreiben
4. Einstellen Plattenkontroller-Register für Schreiben
5. Kopf und Datenschreib-Kommando ausführen
6. Einstellen Map-Register für Lesen
7. Einstellen Plattenkontroller-Register für Lesen
8. Kopf und Datenlese-Kommando ausführen
9. Datenvergleich
10. Wiederholen für alle Sektoren (0-21)

Fehler:

Fehler 1: Statusfehler Schreiben
Fehler 2: Statusfehler Lesen
Fehler 3: Gerät nicht bereit
Fehler 4: Datenvergleichsfehler

Testmethodik:

Die HEAD SELECTION-Logik befindet sich auf der SS-BLP.

6.4.6.22. Test 22: DIFFERENCE LINES TEST Differenzen bei der Zylinderpositionierung

Der Test führt ein Kopf & Datenschreib-Kommando zu den Zylindern 0,1,2,4,8,16,32,64,128,256,512,(PP06), jeweils Spur 0, Sektor 0 durch. Die Daten enthalten die Zylinderadresse. Nach einem RECALIBRATE-Kommando (Nullpositionieren) folgt ein Kopf & Datenlese-Kommando mit den aufgeführten Zylinderadressen und anschließendem Datenvergleich.

Testschritte:

Subtest 1:
1. Initialisieren
2. Nullpositionieren
3. Schreibpuffer füllen
4. Einstellen Plattenkontroller-Register
5. Kopf und Datenschreib-Kommando ausführen
6. Warten auf das Ende der Kommandoausführung
7. Fehlertest
8. Wiederholen mit allen Zylindern

Subtest 2:
1. Initialisieren
2. Nullpositionieren
3. Schreibpuffer füllen
4. Einstellen Plattenkontroller-Register
5. Kopf und Datenlese-Kommando ausführen
6. Warten auf das Ende der Kommandoausführung
7. Fehlertest
8. Datenvergleich
9. Wiederholen mit allen Zylindern

Fehler:

Fehler 1: Statusfehler bei Nullpositionieren
Kopf und Datenschreiben, Kopf und Datenlesen
Fehler 2: Gerät nicht bereit
Fehler 3: Datenvergleichsfehler

Testmethodik:

entfällt;

6.4.6.23. Test 23: SEARCH COMMAND TEST Suchkommando

Es soll ein Such-Kommando nach Zylinder 0, Spur 0, Sektor 0 ausgeführt werden. Zunächst wird jedoch ein Nullpositionier-Kommando angewiesen, um zu sichern, daß sich die Köpfe wirklich auf Zylinder 0 befinden. Dann folgt ein Such-Kommando zu Sektor 0, dem sich ein Datenlese-Kommando zu Sektor 1 anschließt. Die Zeit für diesen Test ist kritisch, da seit dem nur eine Kennzeichnung (Kopfinformation) bis zum Abschluß des Datenlese-Kommandos folgt. Das Lesen darf nicht länger als 1500 Mikrosekunden dauern.

Testschritte:

1. Initialisieren
2. Nullpositionieren
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Such-Kommando für Sektor 0
6. Start des TIMER
7. Datenlese-Kommando für Sektor 1
8. Stop des TIMER

Fehler:

Fehler 1: Zeitüberschreitung (das beabsichtigte
Suchen ist fehlerhaft)
Fehler 2: Statusfehler

Testmethodik:

entfällt;

6.4.6.24. Test 24: SEARCH COMMAND TEST WITH IMPLIED SEEK Suchen einschließlich Positionieren

Es soll ein Such-Kommando nach Zylinder 10, Spur 0, Sektor 0 ausgeführt werden.Zunächst wird jedoch ein Nullpositionier-Kommando angewiesen, um zu sichern, daß sich die Köpfe wirklich auf Zylinder 10 befinden. Dann folgt ein Such-Kommando zu Sektor 0, dem sich ein Datenlese-Kommando zu Sektor 1 anschließt. Die Zeit für diesen Test ist kritisch, da seit dem nur eine Kennzeichnung (Kopfinformation) bis zum Abschluß des Datenlese-Kommandos folgt. Das

Lesen darf nicht länger als 1500 Mikrosekunden dauern.

Testschritte:

1. Initialisieren
2. Nullpositionieren
3. Einstellen Map-Register
4. Einstellen Plattenkontroller-Register
5. Such-Kommando für Sektor 0
6. Start des Timer
7. Datenlese-Kommando für Sektor 0
8. Stop des Timer

Fehler:

Fehler 1: Zeitüberschreitung (das beabsichtigte Suchen ist fehlerhaft)
Fehler 2: Statusfehler

Testmethodik:

entfällt;

6.4.6.25. Test 25: OPERATION INCOMPLETE TEST Operation nicht abgeschlossen

Dieser Test überprüft, daß während eines Lese-Kommandos nicht das OPI-Bit gesetzt wird, falls drei Indeximpulse vorliegen. Ein Datenlese-Kommando wird ab Zylinder 0, Spur 0, Sektor 21 für 46 nachfolgende Sektoren (Bytezähler 23, 525) gestartet. Es wird geprüft, daß das OPI-Bit nicht gesetzt wird bzw. ein anderer Statusfehler vorliegt.

Testschritte:

1. Initialisieren
3. Einstellen Map-Register
   (Map-Transfer innerhalb einer Seite zur Regenerierung des Speichers)
4. Einstellen Plattenkontroller-Register
5. Datenlese-Kommando ausführen
6. Warten auf das Ende der Kommandoausführung
7. Fehlerprüfung

Fehler:

Fehler 1: OPI-Bit gesetzt
Fehler 2: Statusfehler
Fehler 3: Gerät nicht bereit

Testmethodik:

Wenn das OPI-Bit gesetzt ist bedeutet dies, daß 3 Indeximpulse vorgelegen haben, wenn die Erzeugung eines Synchronimpulses ausgeschalten ist. Das OPI-Bit wird auf der SS-BLP erzeugt.

6.4.6.26. Test 26: RECALIBRATE/SEEK TEST Zurück zu Zylinder 0 und Positionieren

Es wird ein Positionier-Kommando zum letzten Zylinder 410 (PP05) bzw. 814 (PP06) angewiesen. Danach wird verglichen, ob der aktuelle Zylinder dem richtigen Wert entspricht und anschließend ein Nullpositionier- Kommando ausgeführt. Weiterhin wird verglichen, ob der aktuelle Zylinder wirklich 0 ist. Die ganze Prozedur wird 10 mal wiederholt.

Testschritte:

1. Initialisieren
2. Positionier-Kommando nach Zylinder 410 (PP05) bzw. 814 (PP06)
3. Prüfen des aktuellen Zylinders (410 oder 814)
4. Fehlerprüfung
5. Nullpositionier-Kommando ausführen
6. Prüfen aktueller Zylinder(0)
7. Fehlerprüfung
8. Wiederholen

Fehler:

Fehler 1: Aktueller Zylinder fehlerhaft
Fehler 2: Statusfehler

Testmethodik:

entfällt;

6.4.6.27. Test 27: SEEK/SEEK TEST Vorwärts/Rückwärts-Positionieren

In diesem Test wird zunächst ein Vorwärts-Positionieren nach Zylinder 128 angewiesen. Nach Vergleich der ordnungsgemäßen aktuellen Zylinderadresse schließt sich ein Rückwärts- Positionieren nach Zylinderadresse 0 an. Nach Überprüfung, ob die aktuelle Zylinderadresse 0 ist, folgt ein allgemeiner Test. Diese Prozedur wird insgesamt 10 mal wiederholt.

Testschritte:

1. Initialisieren
2. Positionieren nach Zylinder 128
3. Prüfen RPCC-Register
4. Prüfen auf andere Statusfehler
5. Prüfen Zeitfaktor
6. Positionieren nach Zylinder 0
7. Prüfen RPCC-Register
8. Prüfen auf andere Statusfehler
9. Prüfen auf Zeitfehler

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: Statusfehler

Fehler 3: RPCC-Register nicht erwartete Stellung

Testmethodik:

entfällt;

6.4.6.28. Test 28: STEPPING SEEK TEST Schrittweises Positionieren

Dieser Test weist Positionier-Kommandos zu den Zylindern 0,1,2, 4,8,16,32,64,128,256,512,(PP06) an, danach wieder rückwärts zu 256,128,64,32 usw. Am Ende jedes Positioniervorganges wird auf Fehler getestet und überprüft, ob das aktuelle Register korrekt ist. Eventuelle Zeitfehler des Gerätes werden mitgeteilt.

Testschritte:

1. Initialisieren
2. RPDC laden
3. Positionier-Kommando auslösen
4. Warten auf Kommandoausführung
5. Fehlertest
6. RPCC-Register prüfen
7. Wiederholen für die restlichen Zylinder

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: Statusfehler
Fehler 3: Inhalt RPCC nicht korrekt

Testmethodik:

entfällt;

6.4.6.29. Test 29: CONVERGING/DIVERGING SEEK TEST Zueinander/voneinander Positionieren

Dieser Test positioniert durch entsprechende Positionier-Kommandos aufeinander zu bzw. voneinander weg. Dazu wird zunächst vom Zylinder 0 zum letzten Zylinder (410 bzw. 814) positioniert. Zu diesen beiden Endpositionen wird jeweils 1 addiert bzw. subtrahiert, bis die Plattenmitte erreicht wird. Danach geht es schrittweise zurück bis wieder die beiden Endpositionen (Zylinder 410 bzw. 814 und Zylinder 0) erreicht sind. Nach jedem Positionier- Kommando wird das aktuelle Zylinderadreß-Register auf den richtigen Wert überprüft.

Testschritte:

1. Initialisieren
2. Voreinstellen ersten und letzten Zylinder
3. Zueinander positionieren, bis erster = letzter Zylinder ist
4. Voneinander positionieren bis erster Zylinder 0 ist

Fehler:

Fehler 1: Aktueller Zylinder ungleich eingestellter Zylinder
Fehler 2: Statusfehler
Fehler 3: Zeitfehler

Testmethodik:

entfällt;

6.4.6.30. Test 30: OFFSET/RETURN TO CENTERLINE TEST Positionieren mit einem Versatz bzw. zurück zur Ausgangsstellung

Der Test überprüft die ordnungsgemäße Arbeit des OFFSET- bzw. des RETURN TO CENTERLINE-Kommandos. Zunächst wird geprüft, ob ein OFFSET SEEK-Kommando mit jedem Versatzwert möglich ist. Anschließend wird überprüft, ob von jedem Versatzwert zur Ausgangsstellung zurückgekehrt werden kann.

Testschritte:

Subtest 1:
1. Initialisieren
2. Versatzadresse einstellen
3. OFFSET SEEK-Kommando ausführen
4. Fehlertest
5. Fortsetzen für alle möglichen Adressen

Subtest 2:
1. Initialisieren
2. Versatzadresse einstellen
3. OFFSET SEEK-Kommando ausführen
4. Fehlertest
5. RETURN TO CENTERLINE.
6. Fehlertest
7. Prüfen, ob Versatzregister 0 ist
8. Fortsetzen für alle möglichen Adressen

Fehler:

Fehler 1: Gerät nicht bereit
Fehler 2: Statusfehler
Fehler 3: Versatzregister ungleich 0 nach RETURN TO CENTERLINE-Kommando

Testmethodik:

entfällt;

6.4.6.31. Test 31: SERVO SETTLE DOWN TEST Servospur vorhanden

Dieser Test prüft, ob die Servospur vorhanden ist und das Gerät einen Zylinder findet (Signal: SEEK COMPLETE). Es wird zwischen Zylinder 0 und 128

positioniert. Ist der Positioniervorgang beendet, wird das LOOK AHEAD-Register gelesen, um zu ermitteln, welcher Sektor sich gerade unter dem Kopf befindet. Hat das Gerät das Positionieren beendet, befindet sich innerhalb der ersten 80% der aktuelle Sektor unter dem Kopf, dann verlangt der Test die Adresse des nächsten Sektor.

Testschritte:

1. Initialisieren
2. Schreibpuffer füllen
3. Einstellen des Map-Registers
4. Wenn RPCC=0, dann zu Zylinder 128 positionieren
5. Wenn RPCC=128, dann zu Zylinder 0 positionieren
6. Prüfen LOOK AHEAD-Register
7. Wenn EXT<1:0>=3 (letztes Fünftel des Sektors unter den Kopf) nächstes Positionieren, sonst nächster Sektor (RPLA+1) nach Adreßregister RPDA
8. Datenschreib-Kommando
9. Warten auf Ende der Kommandoabarbeitung
10. Fehlertest

Fehler:

Fehler 1: Statusfehler
Fehler 2: Zeitfehler des Gerätes

Testmethodik:

Wenn die Positionierrichtung des Gerätes nicht in den Plattenstapel eingeschwenkt wurde oder kein Zylinder zu finden ist, dann wird ein Schreibfehler WRU und die fehlende Kopfauswahl NHS ausgegeben. Fehler, die in diesem Test generell auftreten, deuten daraufhin, daß das Servosystem nicht richtig justiert ist oder daß das Gerät Fehlfunktionen hat oder ein Wechselplattenstapel mit begrenzten Servospuren am Gerät aufliegt.

6.4.6.32. Test 32: PACK ACKNOWLEDGE TEST
Erkennen, verfügbar setzen

Der Test prüft, ob das Verfügbarkeitsbit (VV-Bit) als Ergebnis eines PACK ACKNOWLEDGE-Kommandos gesetzt wird.

Der Test fordert, daß die Netzspannung am Gerät abgeschaltet wird, da nur dadurch das VV-Bit gelöscht werden kann. Zunächst prüft der Test das MOL-Bit. Ist dieses gesetzt, wird der Operator aufgefordert, die Netzspannung für das Gerät abzuschalten. Danach wird der Operator aufgefordert, diese Netzspannung wieder zuzuschalten. Anschließend wird ein PACK ACKNOWLEDGE-Kommando ausgelöst, um zu prüfen, ob das VV-Bit gesetzt wurde.

Testschritte:

1. Initialisieren
2. VV-Bit testen
3. Wenn gesetzt, Operator auffordern, Netzspannung aus und wieder einzuschalten
4. MOL-Bit testen
5. Wenn gelöscht, Operator auffordern, Netz zuzuschalten

6. Prüfen VV-Bit

Fehler:

Fehler 1: MOL-Bit bleibt rückgesetzt
Fehler 2: VV-Bit wird auf Null gehalten
Fehler 3: Ständige Statusfehler

Testmethodik:

entfällt;

6.4.6.33. Test 33: STANDBY COMMAND TEST Entladen, offline setzen

Der Test überprüft, ob das STANDBY-Kommando richtig arbeitet. Zunächst vergleicht der Test, ob das Gerät online ist. Ist dies nicht der Fall, wird der Operator aufgefordert, das Gerät online zu setzen. Der Test prüft dann, ob das MOL-Bit gesetzt ist, um anschließend das STANDBY-Kommando auszulösen. Der Test prüft nun, ob MOL- und VV-Bit gelöscht sind. Der Test fordert nun den Operator auf, diesen Zyklus rückwärts zu durchlaufen, um anschließend zu vergleichen, ob das MOL-Bit gesetzt ist.

Testschritte:

1. Initialisieren
2. Prüfen MOL-Bit
3. Wenn gelöscht, Gerät online setzen (Operatormitteilung)
4. STANDBY-Kommando auslösen
5. MOL-Bit, VV-Bit müssen rückgesetzt sein
6. Starttaste drücken (Operatormitteilung)
7. MOL-Bit prüfen

Fehler:

Fehler 1: MOL-Bit auf 1 gehalten (nach STANDBY-Kommando)
Fehler 2: PIP-Bit auf 0 gehalten (" " ")
Fehler 3: MOL-Bit auf 0 gehalten (nach Restart)

Testmethodik:

entfällt;

6.4.6.34. Test 34: READ IN PRESET COMMAND TEST Anfangsrückstellen, verfügbar setzen

Der Test vergleicht, ob die Platte verfügbar (VV-Bit gesetzt) wird, als Folge eines READ IN PRESET-Kommandos und die Register RPDC, RPDA, RPOF (Bit's ECCI, HCI, FMT22) gelöscht sind. Der Test fordert, daß die Netzspannung des Gerätes abgeschaltet wird, da nur so das VV-Bit gelöscht werden kann. Dazu prüft der Test zunächst das MOL-Bit. Ist dieses gesetzt, dann wird der Operator aufgefordert, die Netzspannung wieder abzuschalten. Danach wird vom Operator gefordert, die Netzspannung wieder einzuschalten. Jetzt wird das READ IN PRESET-Kommando

angewiesen, um anschließend zu prüfen, ob die oben aufgeführten Register gelöscht sind.

Testschritte:

1. Initialisierung
2. VV-Bit testen
3. Wenn gesetzt, Operator auffordern, Netzspannung ab- und wieder einzuschalten
4. MOL-Bit testen
5. Wenn gesetzt, Operator auffordern, Netzspannung ab- und wieder einzuschalten
6. RPDC, RPDA UND RPOF alles 1 schreiben
7. READ IN PRESET-Kommando anweisen
8. VV-Bit prüfen
9. Prüfen RPDC, RPDA UND RPOF, ob gelöscht

Fehler:

Fehler 1: MOL-Bit rückgesetzt
Fehler 2: VV-Bit auf 0 gehalten
Fehler 3: Von den Registern RPDC, RPDA und RPOF wird ein Bit auf 1 gehalten
Fehler 4: Statusfehler

Testmethodik:

entfällt;

6.4.6.35. Test 35: LOGICAL ADDRESS PLUGS TEST Logischer Adreßverbinder

Der Test überprüft die Funktionsfähigkeit der logischen Adreßverbindung. Der Test startet einen Datentransfer im Wartungsmodus. Dann wird der Operator aufgefordert, die logische Adreßverbindung herauszuziehen. Der Test versucht dann einen Zugriff zum Gerätekontroller-Register. Das NED-Bit (nicht existierendes Gerät) müßte im MBA gesetzt sein. Dann wird der Operator angewiesen, die Adreßverbindung wieder einzuschieben. Der Test überprüft, ob das OPE-Bit (Bedienverbindungsfehler) gesetzt ist und Datentransferabbruch im MBA markiert ist, d.h. das VV-Bit ist gelöscht und das Register RPCC auf Null gesetzt (das Gerät ist rückzusetzen).

Testschritte:

1. Prüfen Gerätetyp
2. Wenn nicht PP05 und PP06 - Abbruch
3. Prüfen, ob Gerät online
4. Start Datentransfer
5. Operatoraufforderung Adreßverbindung ziehen
6. Versuch auf RPCS zuzugreifen
7. Prüfen, ob NED-Bit gesetzt
8. Operatoraufforderung Adreßverbindung stecken
9. Prüfen, daß OPE-Bit gesetzt
10. Prüfen, daß VV-Bit gelöscht
11. Prüfen, daß Register RPCC=0

Fehler:

Fehler 1: NED-Bit nicht gesetzt
Fehler 2: OPE-Bit " "
Fehler 3: VV-Bit " gelöscht
Fehler 4: PPCC ungleich 0

Testmethodik:

entfällt;

6.4.6.36. Test 36: WRITE PROTECT SWITCH TEST Schreibschutzschalter

Dieser Test überprüft die Funktionsfähigkeit des Schreibschutzschalters. Der Test schreibt alles "1" auf Zylinder 0, Spur 0, Sektor 0. Dann fordert der Test den Operator auf den Schreibschutzschalter zu setzen. Dann versucht der Test alles "0" auf den gleichen Sektor zu schreiben, um anschließend zu überprüfen, ob das Schreibschutzfehlerbit (WLE-Bit) gesetzt ist. Danach wird der gleiche Sektor gelesen und überprüft, daß die Daten nicht verändert wurden. Anschließend wird dem Operator vom Test angewiesen, den Schreibschutzschalter freizugeben.

Testschritte:

1. Initialisieren
2. Einstellen Map-Register für Schreiben
3. Einstellen Plattenkontroller-Register
4. Schreibpuffer füllen
5. Daten schreiben
6. Schreibschutzschalter setzen
7. Map-Register laden
8. Schreibpuffer mit anderem Muster füllen
9. Versuchen Daten zu schreiben
10. Prüfen, ob WLE-Bit gesetzt
11. Zurücklesen des Sektors
12. Datenvergleich
13. Schreibschutzschalter rücksetzen

Fehler:

Fehler 1: Schreibfehler wenn Schreibschutzschalter nicht gesetzt
Fehler 2: WLE-Bit nicht gesetzt
Fehler 3: Lesefehler
Fehler 4: Datenvergleichsfehler (der Schreibschutz ist nicht wirksam)

Testmethodik:

Wenn das WLE-Bit gesetzt wurde, jedoch alles 0 zurückgelesen wird, arbeitet der Schreibschutzschalter nicht richtig.

6.4.6.37. Test 37: reserviert

6.4.6.38. Test 38: PACK ROTATION SPEED TEST Messen der Umdrehungsgeschwindigkeit

Mit diesem Test wird die Umdrehungsgeschwindigkeit der Platte eines Laufwerkes gemessen. Der Test führt ein Such-Kommando nach Zylinder 0, Spur 0, Sektor 0 aus. Sobald ein Interrupt auftritt, muß der Test unmittelbar noch ein Such-Kommando zum gleichen Sektor ausführen. Das zweite Such-Kommando ist zeitbestimmend. Diese Prozedur wird 10 mal wiederholt und dann der Mittelwert berechnet. Anschließend wird mit einer zulässigen Toleranz verglichen und die beiden Zeiten (der erwartete und der aktuelle) werden ausgegeben.

Die Toleranzen sind: 16,67 ms/Umdrehung, +/-2,9% bei 60Hz
16,67 ms/Umdrehung, +/-2,5% bei 50Hz

Testschritte:

1. Initialisieren
2. Einstellen Plattenkontroler-Register
3. Interrupt erlauben
4. Such-Kommando durchführen
5. Wenn fertig, Start des Timer
6. Such-Kommando durchführen
7. Wenn fertig, Stop des Timer
8. Zeitberechnung und abspeichern
9. Test 10 mal wiederholen
10. Zeit mitteln
11. Zeit prüfen und ausgeben

Fehler:

Fehler 1: Die Zeit überschreitet die Toleranz

Testmethodik:

Wenn die Umdrehungszeit außerhalb der Toleranz liegt, ist zunächst die Netzspannung zu prüfen.Anschließend ist der Sektorzähler zu überprüfen.

6.4.6.39. Test 39: ONE CYLINDER SEEK TIMING TEST Positionierzeit benachbarter Zylinder messen

Der Test führt vorwärts und rückwärts die Positionierung zwischen zwei benachbarten Zylindern durch. Der Test vergleicht, daß die Positionierzeit 10 ms nicht überschreitet.

Testschritte:

1. Initialisieren
2. Vorwärtspositionieren
3. Zeit abspeichern
4. Rückwärtspositionieren
5. Zeit abspeichern
6. Test 100 mal wiederholen

7. Mittlere Positionierzeit berechnen
8. Ausgabe der Positionierzeit

Fehler:

Fehler 1: Die Zeit überschreitet die Toleranz
Fehler 2: Statusfehler (führt zum Testende)

Testmethodik:

Wird die Toleranz der Positionierzeit überschritten, so funktioniert das Servosystem nicht.

6.4.6.40. Test 40: ACCESS TIME MEASUREMENT TEST
Messen der Zugriffszeit

Dieser Test positioniert vorwärts und rückwärts zwischen Zylinder 0 und 136. Der Test vergleicht, daß die Positionierzeit nicht 35 ms überschreitet.

Testschritte:

1. Initialisieren
2. Vorwärtspositionieren
3. Zeit registrieren
4. Rückwärtspositionieren
5. Zeit registrieren
6. Test 100 mal wiederholen
7. Durchschnittliche Positionierzeit berechnen
8. Ausgabe der berechneten Zeiten

Fehler:

Fehler 1: Die Zeit überschreitet die Toleranz
Fehler 2: Statusfehler (führt zum Testende)

Testmethodik:

Wenn die Positionierzeiten die Toleranzgrenzen überschreiten, funktioniert das Servosystem nicht.

6.4.6.41. Test 41: MAXIMUM SEEK TIMING TEST
Maximale Positionierzeit

Dieser Test positioniert vorwärts und rückwärts zwischen Zylinder 0 und 410 (PP05) bzw. 814 (PP06). Der Test prüft, daß die Positionierzeit 60 ms nicht überschreitet.

Testschritte:

1. Initialisieren
2. Vorwärtspositionieren
3. Zeit registrieren
4. Rückwärtspositionieren
5. Zeit registrieren

6. Test 100 mal wiederholen
7. Durchschnittliche Positionierzeit berechnen
8. Ausgabe der berechneten Zeiten

Fehler:

Fehler 1: Die Zeit überschreitet die Toleranz
Fehler 2: Statusfehler (führt zum Testende)

Testmethodik:

Wenn die Positionierzeiten die Toleranzgrenzen überschreiten, funktioniert das Servosystem nicht.

6.4.7. Auflistung der Tests

Test 1: NOOP COMMAND
Keine Operation
Test 2: DRIVE CLEAR
Gerät rücksetzen
Test 3: WRITE HEADER & DATA, READ HEADER & DATA TESTS
Schreiben Kopf & Daten, Lesen Kopf & Daten
Test 4: WRITE/READ DATA TESTS
Daten Schreiben/Lesen
Test 5: WRITE/READ DATA TESTS
Daten Schreiben/Lesen
Test 6: WRITE/READ DATA TESTS
Daten Schreiben/Lesen
Test 7: WRITE/READ DATA TESTS
Daten Schreiben/Lesen
Test 8: IMPLIED SEARCH TEST
Datentransfer einschließlich Suchen
Test 9: IMPLIED SEEK TEST
Datentransfer einschließlich Positionieren
Test 10: SEEK AND WRITE TEST (CYL 0-10)
Positionieren und Schreiben
Test 11: INTERRUPT TEST
Interrupterkennung
Test 12: WRITE CHECK HEADER & DATA TEST
Schreiben Kopf und Daten mit Prüfung
Test 13: WRITE CHECK DATA TEST
Daten schreiben mit Prüfung
Test 14: LAST SECTOR TRANSFERED 'LST' TEST
Datentransfer mit dem letzten Sektor
Test 15: ADDRESS OVERFLOW ERROR 'AOE' TEST
Fehlerprovokation Adreßüberlauf
Test 16: INVALID ADDRESS ERROR TEST
Fehlerprovokation falsche Adresse
Test 17: FORMAT ERROR TEST
Fehlerprovokation falsches Format
Test 18: ILLEGAL FUNCTION TEST
Fehlerprovokation illegale Funktion
Test 19: HEADER COMPARE ERROR
Fehlerprovokation falsche Kopfinformation
Test 20: HEAD SELECTION TEST
Spurauswahl aus der Kopfinformation

Test 21:	SECTOR SELECTION TEST Sektorauswahl aus der Kopfinformation
Test 22:	DIFFERENCE LINES TEST bei der Zylinderpositionierung
Test 23:	SEARCH COMMAND Suchkommando
Test 24:	SEARCH COMMAND TEST, WITH IMPLIED SEEK Suchen einschließlich Positionieren
Test 25:	OPERATION COMPLETE TEST Operation nicht abgeschlossen
Test 26:	RECALIBRATE/SEEK TEST Zurück zu Zylinder 0 und Positionieren
Test 27:	SEEK/SEEK TEST Vorwärts/Rückwärts-Positionieren
Test 28:	STEPPING SEEK TEST Schrittweises Positionieren
Test 29:	CONVERGING/DIVERGING SEEK TEST Zueinander/voneinander Positionieren
Test 30:	OFFSET/RETURN TO CENTERLINE TESTS Positionieren mit einem Versatz bzw. zurück zur Ausgangsstellung
Test 31:	SERVO SETTLE DOWN TEST Servospur vorhanden
Test 32:	PACK ACKNOWLEDGE TEST Erkennen, verfügbar setzen
Test 33:	STANDBY COMMAND TEST Entladen, offline setzen
Test 34:	READ IN PRESET COMMAND TEST Anfangsrückstellen, verfügbar setzen
Test 35:	LOGICAL ADDRESS PLUGS TEST Logischer Adreßverbinder
Test 36:	WRITE PROTECT SWITCH TEST Schreibschutzschalter
Test 37:	reserviert
Test 38:	PACK ROTATION SPEED TEST Messen der Umdrehungsgeschwindigkeit
Test 39:	ONE CYLINDER SEEK TIMING TEST Positionierzeit benachbarter Zylinder
Test 40:	ACCESS TIME MEASUREMENT TEST Messen der Zugriffszeit
Test 41:	MAXIMUM SEEK TIMING TEST Maximale Positionierzeit

6.5. PVRCA: Kontrollerdiagnose ohne Datenträger für PP05, PP06

6.5.1. Zusammenfassung

Das Programm PVRCA (Dok.-Nr. 1.57.700605.6) ist ein Diagnoseprogramm der Ebene 3 (Stand-alone-Modus). Das Programm besteht aus einer Testserie, welche die ordnungsgemäße Beschaffenheit der Wechselplattenspeicher CM 5404 bzw. CM 5416 nachweist. Das Programm enthält eine Reihe von Funktionstests mit folgendem Inhalt:

1) Register - Tests
2) Kontroller-Tests
3) Datentransfer-Tests
4) Fehlerbit-Funktions-Tests

6.5.2. Testbedingungen

6.5.2.1. Hardware

- RVS K 1840 (mit Minimalkonfiguration)
- MSBUS-Adapter K 2816
- Wechselplattenspeicher PP05, PP06 (CM 5404, CM 5416)

- Datenträger
  Das Programm fordert, daß die Platte nicht aufliegt (wird entsprechend simuliert). Wenn das Testprogramm eine Platte findet, die aufliegt, wird der Test sofort abgebrochen.

6.5.2.2. Software

- Diagnose-Supervisor, über BOOT-Kommando gestartet.

6.5.3. Voraussetzungen

Der Test setzt voraus, daß der MSBUS-Adapter arbeitsfähig ist. Das heißt, das Testprogramm PVCAA muß fehlerfrei gelaufen sein.

6.5.4. Bedienungshinweise

Nachdem der Diagnose-Supervisor über BOOT-Kommando geladen und gestartet wurde, sind Kontroller und Gerät anzuschließen.

```
DS> ATTACH RHA40            ; Anschließen MSBUS-Adapter
DEVICE LINK? HUB            ;
DEVICE NAME? RHO            ;
TR? 8                       ;
BR? 5                       ;
DS> ATTACH PP05             ; Anschließen Wechselplattenspeicher
```

```
DEVICE LINK? RHO          ;
DEVICE NAME? DBA0         ;
DS> SELECT DBA0           ; Gerät auswählen
DS> LOAD PVRCA            ;
DS> START                 ;
```

6.5.5. Funktionsbeschreibung

6.5.5.1. Übersicht

PVRCA läuft unter dem Diagnose-Supervisor und paßt sich den RVS- Diagnosebedingungen an. Es hat einen Kopfmodul, welcher Initialisierungs- CLEAN-UP und eine Reihe anderer Subroutinen enthält. Außerdem sind hier alle globalen Text-, Daten- und Puffervereinbarungen enthalten.

```
Ein Testmodul enthält die Register-Tests            Test 1  ... 11
 "     "        "     die Kontroller-Kommando-Tests Test 12 ... 19
 "     "        "     die Datentransfer-Tests       Test 20 ... 36
 "     "        "     die Fehlerbit-Funktions-Tests Test 37 ... 42
 "     "        "     die Fehlerbit-Funktions-Tests Test 43 ... 47
```

6.5.5.2. Programmgröße

Speicherbelegung von 200(x) ... B5FF(x), das sind B400(x), d.h. 46080. byte.

6.5.5.3. Programmlaufzeit

Insgesamt ca. 3 min

6.5.5.4. Abarbeitungsbedingungen

Der Diagnose-Supervisor beginnt den Programmablauf immer mit der Initialisierungsroutine und schließt es mit der CLEAN-UP-Routine ab.

6.5.5.5. Flagverwendung

entfällt

6.5.5.6. Allgemeine Programmanforderungen

entfällt

6.5.5.7. Fehlermitteilung

1. Datenvergleichsfehler

```
** PVRCA PPOX/DCL DIAGNOSTIC - 1.0 **

ADDRESS          GOOD    BAD    XOR

0000AE08:        55      00     55              ; BIT6,BIT4,BIT2,BIT0
0000AE09:        55      00     55              ; BIT6,BIT4,BIT2,BIT0
0000AE0A:        55      00     55              ; "
0000AE0B:        55      00     55              ; "
0000AE0C:        55      00     55              ; "
0000AE0D:        55      00     55              ; "
0000AE0E:        55      00     55              ; "
0000AE0F:        55      00     55              ; "
0000AE10:        55      00     55              ; "
0000AE11:        55      00     55              ; "
```

2. Fehlerbitauswertung

```
DS> ST/TE:37
  Programm: PVRCA PP0X/DCL DIAGNOSTIC, revision 1.0, 47 tests, at 09:03:37.04.
Testing: _DBA0
Test 37: ATTENTION ACTIVE BIT (ATA)
Test 38: FORMAT ERROR (FER) BIT TEST
Test 39: HEADER COMPARE ERROR (HCE) TEST

*** PVRCA PP0X/DCL DIAGNOSTIC 1.0 ***

Pass 1, test 39, subtest 1, error 4, 12-MAY-1987 09:03:38.83
Hard error while testing DBA0: EXCEPTION STUCK @ ZERO

*** End of Hard error number 4 ***

MBA CHANNEL STATUS DUMP
MBA_CSR:   [00F28000]   00000000(X)       ;
MBA_CR :   [00F28004]   00000000(X)       ;
MBA_SR :   [00F28008]   00013040(X)       ; ATTN, DTCOMP,DTABT,MDPE
MBA_VAR:   [00F2800C]   00000002(X)       ;
MBA_MAP(40):            00000000(X)       ;
MBA_BCNT:  [00F28010]   FDFCFDFA(X)       ;
```

3. Schreibfehler

```
DS> ST/TE:34
  Programm: PVRCA PP0X/DCL DIAGNOSTIC, revision 1.0, 47 tests, at 09:16:36.11.
Testing: _DBA0

Test 34: WRITE CHECK DATA (MAINT MODE) DATA=00
*** PVRCA PPOX/DCL DIAGNOSTIC - 1.0 ***
Pass 1, test 34, subtest 0,error 5, 12-MAY-1987 09:16:37.37
Hard error while testing DBA0: ERROR DETECTED DURING WRITE CHECK
                               DATA COMMAND
*** End of Hard error number 5 ***

Register              Daten                Bezeichnung
RPCS1  [40F28400]:    0828(X)            ; DVA,FUNCTION=WRITE CHECK DATA
```

```
RPDS   [40F28404]:   01C0(X)        ; DPR, DRY, VV
RPER1  [40F28408]:   0000(X)        ;
RPMR   [40F2840C]:   0101(X)        ; ZDT, DMD
RPAS   [40F28410]:   0000(X)        ;
RPDA   [40F28414]:   0102(X)        ; TRACK=01(D), SECTOR=02(D)
RPDT   [40F28418]:   2012(X)        ; MOH, DRIVE TYPE=PP06
RPLA   [40F2841C]:   0070(X)        ; SECTOR=01(D), EXT1, EXT0
RPSN   [40F28420]:   0000(X)        ;
RPOF   [40F28424]:   9800(X)        ; SGCH,EMT22,ECCI
RPDC   [40F28428]:   0000(X)        ; DESIRED CYLINDER=0000(D)
RPCC   [40F2842C]:   0000(X)        ; CURRENT CYLINDER=0000(D)
RPER2  [40F28430]:   0000(X)        ;
RPER3  [40F28434]:   0000(X)        ;
RPEC1  [40F28438]:   0000(X)        ;
RPEC2  [40F2843C]:   0000(X)        ;

MBA CHANNEL STATUS DUMP

MBA_CSR:    [00F28000]   00000000(X)   ;
MBA_CR :    [00F28004]   00000000(X)   ;
MBA_SR :    [00F28008]   00003040(X)   ; DTCOMP,DTABT, MDPE
MBA_VAR:    [00F2800C]   00000026(X)   ;
MBA_MAP(40):             00000000(X)   ;
MBA_BCNT:   [00F28010]   FFC2FFE2(X)   ;
```

6.5.5.8. Programmitteilungen

entfällt

6.5.5.9. Leistungsparameter des Testobjektes

entfällt

6.5.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.5.5.11. Testfolge bei mehreren Einheiten

Bei mehreren Einheiten werden sämtliche ausgewählten Tests zunächst für eine Einheit abgearbeitet. Im Anschluß daran die gleiche Testfolge für die nächste Einheit. Die Auswahl der Einheiten erfolgt in der Reihenfolge der Selektierung.

6.5.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.5.5.13. Testsektionen

PVRCA PPOX FUNCTIONAL DIAGNOSTIC enthält folgende Sektionen:
DEFAULT, REGISTER, COMMAND, DATA_XFER, ERROR, ECC_TEST

Beschreibung der Testsektionen:

1) DEFAULT - Standardtestsektion
   Hier sind alle Tests enthalten, also Test 1-47;

2) REGISTER - Registersektion
   Test 1-11

Diese Tests versuchen folgende Datenmuster in jedes Schreib/Lese-Register des Plattenkontrollers zu schreiben:
alles Null, alles Eins, durchlaufende Null, durchlaufende Eins.
Danach wird jedes Register gelesen, um zu vergleichen, ob die richtigen Bits gesetzt sind. Diesen Tests liegt folgende Methodik zugrunde:
Durch die Registertests werden die benötigten Teile des MSBUS-Kontrollers und die Registerlogik des Plattenkontrollers geprüft. Läuft nur einer dieser Tests fehlerhaft, dann liegt der Fehler wahrscheinlich am jeweiligen Register selbst. Sind aber mehrere Tests fehlerhaft, dann kann der Fehler an der Dual-Port-BLP, dem MSBUS-Interface oder am Kabel liegen. Wird das Problem im Kabel oder MSBUS-Interface vermutet, sollte der Test mit einem anderen Gerät durchgeführt werden. Bleibt der Fehler weiterhin bestehen, so handelt es sich dann wahrscheinlich doch um einen Fehler des MBA bzw. des Kabels. Es ist zu raten, daß die Art der Probleme, welche die Tests aufdecken, sich am besten durch Einkreisen lösen lassen. Jeder Test beschreibt, wo die entsprechende Logik in der Dokumentation zum Plattenkontroller zu finden ist.

3) COMMAND - Kontroller-Kommandotestsektion
   Test 12 - 19

Diese Sektion testet die HOUSEKEEPING (sich nur auf den Kontroller beziehende)-Kommandos und Positionierkommandos bei Nutzung des besonderen Wartungsmodus des Plattenkontrollers. Diese Tests setzen voraus, daß in den Registern des Plattenkontrollers Test 1-10 erfolgreich gelaufen sind, und daß das MSBUS_INIT-Signal ordnungsgemäß arbeitet.

4) DATA_XFER - Datentransfertestsektion
   Test 20 - 36

Die ersten Tests dieser Sektion testet die Funktion Kopf & Daten schreiben, die unter dem Gesichtspunkt des Wartungsmodus bezüglich des Plattenkontrollers genutzt werden. Das Modul enthält 6 Tests mit dem Kopf & Daten-Schreibkommando mit folgenden Datenmustern:

1) alles 0
2) alles 1
3) AA(X)
4) 55(X)
5) durchlaufende 1
6) durchlaufende 0

Die nächsten Tests entsprechen den vorangegangenen, aber mit Kopf & Daten-Lesekommando.

Die nachfolgenden Tests prüfen die Funktion Daten schreiben, Daten lesen, Schreiben mit Prüfung, Schreiben Kopf mit Prüfung und Sektorauswahl, die unter dem Gesichtspunkt des Wartungsmodus bezüglich des Plattenkontrollers genutzt werden.

5) ERROR - Fehlerbittestsektion
Test 37 - 47

Diese Sektion enthält die Funktionstests für Fehlerbits. Durch softwaremäßige Kommandofolgen wird versucht, die jeweiligen Fehlerbedingungen nachzuempfinden.

6) ECC_TEST - reserviert

6.5.5.14. Verschiedenes

entfällt

6.5.6. Testbeschreibungen

6.5.6.1. Test 1: SUBSYSTEM REGISTER ADDRESS TEST Subsystem Registeradressierung

Der Zweck dieses Tests besteht darin, die richtige Kommunikation zwischen CPU und den MBA sicherzustellen. Dies ist erfüllt, wenn jedes Kanal/Geräte-Register gelesen werden kann.

Testschritte:

1. MBA initialisieren
2. MBA Status lesen
3. Zugriff zu den Kanal/Geräte-Registern
4. Prüfen auf Zugriffsfehler der Register
5. Abbruch, wenn auf das Register nicht zugegriffen werden kann
6. Wiederholen der vorangegangenen Testschritte für alle Kanal/Geräte-Register auf die zugegriffen werden kann

Testmethodik:

Bringt der Test Fehler, so ist dieser in der Logik der Adreßdekodierung des Kanal/Geräte-Registers zu lokalisieren.

6.5.6.2. Test 2: QUALIFICATION TEST Qualifikation

Dieser Test überprüft, ob der MSBUS in Gang gesetzt ist, das Gerät für den Test existiert, verfügbar (DVA- und DRY-Bit gesetzt) und offline ist (MOL-Bit rückgesetzt).

Testmethodik:

entfällt

6.5.6.3. Test 3: RPCS1 REGISTER RPCS1 (Statuskommando)-Register

Der Test prüft das RPCS1-Register. Es werden nur Bit 1 bis 5 geschrieben, da nur diese Lese/Schreib-Bits sind.

Testmethodik:

Die RPCS1-Logik ist auf der RG-BLP enthalten. Der Fehler ist entweder auf der RG-BLP oder auf der DP-BLP oder möglicherweise im MSBUS-Interface zu suchen. Der beste Weg das festgehaltene Bit zu isolieren ist es herauszufinden, ob das gleiche Bit in einem anderen Plattenkontrollerregister ebenfalls gehalten wird oder nicht. Wenn ja liegt das Problem wahrscheinlich in der DP-BLP oder im MSBUS. Es ist klar, daß diese Problemart sich am besten durch Einkreisen des festgehaltenen Bit's lösen läßt.

6.5.6.4. Test 4: RPDA - REGISTER RPDA (Diskadreß) - Register

Dieser Test prüft das RPDA-Register. Nur die Bits 0 bis 4 und 8 bis 12 werden geschrieben bzw. gelesen.

Testmethodik:

Die Logik des RPDA-Registers befindet sich auf der SS-BLP. Werden alle Bits des Registers festgehalten wird wahrscheinlich das entsprechende Auswahlsignal für den Registerzugriff auf 1 gehalten. Man prüfe, ob nur dieses Register fehlerhaft ist. Ist dies der Fall, dann handelt es sich um ein Problem der SS-BLP, wenn nicht dann ist in der DP-BLP oder MSBUS-Interface nachzusehen.

6.5.6.5. Test 5: RPER1 - REGISTER RPER1 (Fehler) - Register

Dieser Test überprüft, ob beim Fehlerregister 1 ein Bit auf 0 oder 1 gehalten wird.

Testmethodik:

Die Logik des RPER1-Registers befindet sich auf der RG-BLP. Wenn alle Bits des Registers nicht schalten, dann prüfe man die entsprechenden Auswahlsignale für den Registerzugriff. Wenn ein Bit auf 1 gehalten wird, so prüfe man das direkte Setzsignal des Fehler Flip-Flop. An diesem Punkt des Diagnoseprogrammes (angenommen der vorangegangene Test ist durchgelaufen), muß man das MSBUS-Interface bzw. die DP-BLP in die Fehlersuche einbeziehen.

6.5.6.6. Test 6: RPMR REGISTER RPMR (Wartungs) - Register

Dieser Test überprüft, ob im RPMR(MAINTENANCE)-Register Bits auf 1 oder 0 gehalten werden. Es muß zunächst Bit 0 gesetzt werden, bevor auf die restlichen Bits zugegriffen werden kann.

Testmethodik:

Bit 0 und 4 sind lese- und schreibbar, der Rest nur lesbar. Dieser Test ist nur ein Test der Schreib/Lese-Bits. Bit 0 (Diagnose-Modus) erlaubt die anderen Bits zu lesen bzw. zu schreiben. Wenn Bit 0 nicht gesetzt ist, werden die anderen Bits immer nur als 0 zurückgelesen. Wird eines von diesen Bits festgehalten, überprüfe man das entsprechende Setzsignal. So können die Fehler der Schreib/Lese-Bits des Wartungsregisters gefunden werden. Die nur Lese-Bits werden später geprüft, wenn im Wartungsmodus geschrieben und gelesen wird.

6.5.6.7. Test 7: RPOF - REGISTER
RPOF (Versatz) - Register

Der Test prüft, ob im RPOF(OFFSET)-Register Bits auf 0 oder 1 gehalten werden.

Testmethodik:

Die OFFSET-Registerlogik befindet sich auf der RG-BLP. Wenn alle Bit festgehalten werden, dann prüfe man die entsprechenden Auswahlsignale für den Registerzugriff.

6.5.6.8. Test 8: RPDC - REGISTER
RPDC(Soll-Zylinderadreß)-Register

Dieser Test prüft, ob im RPDC(DESIRED CYLINDER ADDRESS)-Register Bits auf 0 oder 1 gehalten werden. Bei der Prüfung bestimmt der Gerätetyp (PP05, PP06), wieviele DCA-Bits im Register verwendet werden.

Testmethodik:

Die Logik des RPDC-Registers findet man auf der SS-BLP. Falls alle Bits festgehalten werden, prüfe man die entsprechenden Auswahlsignale für den Registerzugriff. Weiterhin prüfe man das Löschsignal für das Register.

6.5.6.9. Test 9: RPER2 - REGISTER
RPER2 (Fehler 2)- Register

Dieser Test prüft, ob im ER2 (Fehler 2)-Register Bits auf 0 oder 1 gehalten wewrden.

Testmethodik:

Die Logik des Registers befindet sich auf der EC-BLP. Wird ein Bit des Registers auf 1 gehalten, folgt daraus ein Hardwarefehler im Gerät. Man prüfe die direkten Setzeingänge des Flip-Flops, um das Setzereignis nachzuweisen. Dieser Test prüft nur die aktuelle Registerbelegung, jedoch nicht die Fehlerentschlüsselungslogik. Werden alle Bits auf 0 gehalten, so prüfe man die entsprechenden Auswahlsignale für den Registerzugriff.

6.5.6.10. Test 10: RPER3 REGISTER
RPER3 (Fehler 3)-Register

Dieser Test prüft, ob im RPER3 - Register Bits auf 0 oder 1 gehalten werden.

Testmethodik:

Die Logik des RPER3-Registers befindet sich auf der EC-BLP. Wird ein Bit auf 1 gehalten, führt dies zu einer Hardwarefehlerbedingung des Gerätes. Man prüfe die direkten Setzeingänge des Flip-Flop. Werden alle Bits auf 0 gehalten, so prüfe man die entsprechenden Auswahlsignale für den Registerzugriff.

6.5.6.11. Test 11: MBA INIT
MBA Initialisierung

Dieser Test schreibt alles 1 in jedes Schreib/Lese-Register. Anschließend wird der MSBUS initialisiert. Es wird geprüft, ob danach alle Bits gelöscht sind.

Testmethodik:

Das INIT-Signal greift in den Plattenkontroller über ein entsprechendes Signal auf der DP-BLP ein. Im Plattenkontroller wird damit ein generelles Rücksetzen angeregt. Der beste Weg den Fehler zu finden, ist die Beschaltung des fehlerhaften Registers rückwärts zu verfolgen, um den fehlerbehafteten Eingangskreis zu ermitteln.

6.5.6.12. Test 12: DRIVE CLEAR
Gerät rücksetzen

Dieser Test schreibt alles 1 in jedes Schreib/Lese-Register und weist danach das Kommando "Gerät rücksetzen" an. Es wird überprüft, daß alle Bits gelöscht sind.

Testmethodik:

Das Kommando "Gerät rücksetzen" wird auf der RG-BLP dekodiert. Wenn der Test MBA INIT fehlerfrei läuft, dann wird das Kommando "Gerät rücksetzen" wahrscheinlich nicht richtig dekodiert, da die Kommandos "MBA initialisieren" und "Gerät rücksetzen" teilweise die gleiche Logik benutzen.

6.5.6.13. Test 13: PACK ACKNOWLEDGE AND READ IN PRESET COMMAND
Erkennungskommando, Anfangsrückstellen

Dieser Test überprüft das Erkennungskommando und die Anfangsrückstellung. Es wird also das VV-Bit (Einheit verfügbar) im Gerätestatusregister überprüft. Das Erkennungskommando muß das VV-Bit setzen. Das Kommando zur Anfangsrückstellung muß das VV-Bit setzen, RPDC, RPDA und RPOF (Bit: FMT22, ECC, HCI) löschen. Kann das VV-Bit nicht gesetzt werden, wird das Diagnoseprogramm abgebrochen. Der Grund für den Abbruch ist, daß das VV-Bit die Dekodierung sämtlicher Positionier- und Schreib/Lese-Kommandos erlaubt.

Testmethodik:

Die Dekodierlogik beider Kommandos befindet sich auf der DP-BLP. Bei einem fehlerhaften Test überprüft man zunächst die richtige Dekodierung der Kommandos, das Setzen des VV-Bits und zum Schluß, daß die richtigen Auswahlsignale für die RPDA- und RPDC-Register (siehe SS-BLP) anliegen.

6.5.6.14. Test 14: NOOP COMMAND Keine Operation

Dieser Test prüft die ordnungsgemäße Arbeit des Kommandos "keine Operation". Nach Ausführung des Kommandos im Wartungsmodus werden alle Register geprüft.

Testmethodik:

Dieser Test prüft, ob sich der Inhalt der Register verändert hat, nachdem das Kommando "keine Operation" für den Plattenkontroller ausgeführt wurde. Wenn sich ein Register nach dieser Operation verändert, dann wird die Funktion wahrscheinlich nicht richtig dekodiert. Man sollte die Logik des Funktionsdekoders auf der RG-BLP überprüfen. Eine andere Möglichkeit wäre, die fehlerhafte Arbeit der Registerauswahllogik auf der gleichen BLP bzw. auf der Interfacekontroll-BLP zu überprüfen.

6.5.6.15. Test 15: SEEK COMMAND Positionierkommando

Der Test prüft die Dekodierlogik für das Positionierkommando. Nach der MBA-Initialisierung folgt das Setzen des Wartungsmodus und schließlich das Positionierkommando. Der Test überprüft, ob das DRY-Bit gelöscht und das PIP-Bit gesetzt sind.

Testmethodik:

Die Logik zur Dekodierung des Positionierkommandos befindet sich auf der RG-BLP. Das DRY-Bit ist praktisch die Negation des GO-Bit, so daß für das DRY-Bit die entsprechenden Setzsignale des GO-Bit überprüft werden sollten.

6.5.6.16. Test 16: RECALIBRATE COMMAND Nullpositionierung

Dieser Test prüft die ordnungsgemäße Dekodierung des Kommandos zur Positionierung auf Zylinder 0. Die Nullpositionierung wird im Wartungsmodus angewiesen. Dann wird überprüft, ob das DRY-Bit gelöscht und PIP- und GO-Bit gesetzt sind.

Testmethodik:

Die Dekoderlogik für dieses Kommando befindet sich auf der RG-BLP. Für einen Fehler ist vermutlich diese BLP verantwortlich.

6.5.6.17. Test 17: OFFSET COMMAND Versatzkommando

Dieser Test prüft die Dekodierlogik für das Versatzpositionierkommando. Der Test erwartet, daß das DRY-Bit gelöscht und GO- und PIP-Bit gesetzt sind. Bevor das Kommando dekodiert werden kann, muß das VV-Bit gesetzt sein.

Testmethodik:

Die Logik zur Dekodierung des Positionierkommandos befindet sich aufder RG der RG-BLP. Das DRY-Bit ist praktisch die Negation des GO-Bit, so daß für das DRY-Bit die entsprechenden Setzsignale des GO-Bit überprüft werden sollten.

6.5.6.18. Test 18: RETURN TO CENTERLINE Zurück zur Ausgangsposition

Dieser Test überprüft die Dekodierlogik für das Kommando "zurück zur Ausgangsposition".

Testmethodik:

Die Logik zur Dekodierung des Positionierkommandos befindet sich auf der RG-BLP. Das DRY-Bit ist praktisch die Negation des GO-Bit, so daß für das DRY-Bit die entsprechenden Setzsignale des GO-Bit überprüft werden sollten.

6.5.6.19. Test 19: SEARCH COMMAND & LOOK AHEAD REGISTER Suchkommando und Vorschauregister

Dieser Test prüft die Sektorzählerlogik und das LOOK AHEAD-Register. Der Test weist das Suchkommando im Wartungsmodus nach Zylinder 0, Spur 0, Sektor 21 an. Das LOOK AHEAD-Register wird nach dem Indeximpuls geprüft (muß 0 sein). Das Bereichsfeld wird in jedem Sektor nach 128 byte, dann nach weiteren 128 byte und dann nach 256 byte geprüft. Der Sektorzähler wird nach jedem Sektor geprüft. Die Logik des Sektorzählers ist auf der DP-BLP. Hier sind bei Fehler die entsprechenden Setz- und Löschsignale zu überprüfen.

Testmethodik:

entfällt;

6.5.6.20. Test 20: WRITE HEADER & DATA (MAINT. MODE) DATA = 00 Kopf & Daten schreiben (Wartungsmodus) Datenmuster 00(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	=	0
Spur	=	0
Sektor	=	0
Schlüsselnummer	=	0

Bytezähler = 520
Datenmuster = 00(X)

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer mit alles 1 füllen
4. MAP-Register des MBA (Datenkanal) einstellen
5. Einstellen RPCS1-Register
6. Setzen ECC INHIBIT-Bit im RPOF-Register
7. Setzen FMT22-Bit im RPOF-Register
8. Setzen Wartungsmodus
9. Setzen GO-Bit im RPCS1-Register
10. Ansprung SEARCH(Such)-Routine
11. Ansprung Kopf & Daten-Schreibroutine
12. Prüfen auf Kanalfehler
13. Prüfen auf Statusfehler (RPDS)
14. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.21. Test 21: WRITE HEADER & DATA (MAINT. MODE)
DATA = FF
Kopf & Daten schreiben (Wartungsmodus) Datenmuster FF(X)

Das Kommando wird mit folgenden Parametern ausgeführt:
Zylinder = 0
Spur = 0
Sektor = 0
Schlüsselnummer = 0
Bytezähler = 520
Datenmuster = FF(X)

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer füllen
4. Plattensimulationspuffer mit alles 1 füllen
5. MAP-Register des MBA (Datenkanal) einstellen
6. Einstellen RPCS1-Register
7. Setzen ECC INHIBIT-Bit im RPOF-Register
8. Setzen FMT22-Bit im RPOF-Register
9. Setzen Wartungsmodus
10. Setzen GO-Bit im RPCS1-Register
11. Ansprung SEARCH(Such)-Routine
12. Ansprung Kopf & Daten-Schreibroutine
13. Prüfen auf Kanalfehler
14. Prüfen auf Statusfehler (RPDS)
15. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.22. Test 22: WRITE HEADER & DATA (MAINT. MODE)
DATA = AA
Kopf & Daten schreiben (Wartungs-modus) Datenmuster AA(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	= 0
Spur	= 0
Sektor	= 0
Schlüsselnummer	= 0
Bytezähler	= 520
Datenmuster	= AA(X)

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer füllen
4. Plattensimulationspuffer mit alles 1 füllen
5. MAP-Register des MBA (Datenkanal) einstellen
6. Einstellen RPCS1-Register
7. Setzen ECC INHIBIT-Bit im RPOF-Register
8. Setzen FMT22-Bit im RPOF-Register
9. Setzen Wartungsmodus
10. Setzen GO-Bit im RPCS1-Register
11. Ansprung SEARCH(Such)-Routine
12. Ansprung Kopf & Daten-Schreibroutine
13. Prüfen auf Kanalfehler
14. Prüfen auf Statusfehler (RPDS)
15. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.23. Test 23: WRITE HEADER & DATA (MAINT. MODE)
DATA = 55
Kopf & Daten schreiben (Wartungs-modus) Datenmuster 55(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	= 0
Spur	= 0
Sektor	= 0
Schlüsselnummer	= 0
Bytezähler	= 520
Datenmuster	= 55(X)

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer füllen
4. Plattensimulationspuffer mit alles 1 füllen
5. MAP-Register des MBA (Datenkanal) einstellen
6. Einstellen RPCS1-Register
7. Setzen ECC INHIBIT-Bit im RPOF-Register
8. Setzen FMT22-Bit im RPOF-Register
9. Setzen Wartungsmodus
10. Setzen GO-Bit im RPCS1-Register
11. Ansprung SEARCH(Such)-Routine
12. Ansprung Kopf & Daten-Schreibroutine
13. Prüfen auf Kanalfehler
14. Prüfen auf Statusfehler (RPDS)
15. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.24. Test 24: WRITE HEADER & DATA (MAINT. MODE) FLOATING DATA 1 Kopf & Daten schreiben (Wartungsmodus) Datenmuster durchlaufende 1

Zylinder	=	0
Spur	=	0
Sektor	=	0
Schlüsselnummer	=	0
Bytezähler	=	520
Datenmuster	=	durchlaufende 1

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer füllen
4. Plattensimulationspuffer mit alles 1 füllen
5. MAP-Register des MBA (Datenkanal) einstellen
6. Einstellen RPCS1-Register
7. Setzen ECC INHIBIT-Bit im RPOF-Register
8. Setzen FMT22-Bit im RPOF-Register
9. Setzen Wartungsmodus
10. Setzen GO-Bit im RPCS1-Register
11. Ansprung SEARCH(Such)-Routine
12. Ansprung Kopf & Daten-Schreibroutine
13. Prüfen auf Kanalfehler
14. Prüfen auf Statusfehler (RPDS)
15. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.25. Test 25: WRITE HEADER & DATA (MAINT. MODE)
FLOATING DATA 0
Kopf & Daten schreiben (Wartungsmodus) Datenmuster durchlaufende 0

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	=	0
Spur	=	0
Sektor	=	0
Schlüsselnummer	=	0
Bytezähler	=	520
Datenmuster	=	durchlaufende 0

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Schreibpuffer füllen
4. Plattensimulationspuffer mit alles 1 füllen
5. MAP-Register des MBA (Datenkanal) einstellen
6. Einstellen RPCS1-Register
7. Setzen ECC INHIBIT-Bit im RPOF-Register
8. Setzen FMT22-Bit im RPOF-Register
9. Setzen Wartungsmodus
10. Setzen GO-Bit im RPCS1-Register
11. Ansprung SEARCH(Such)-Routine
12. Ansprung Kopf & Daten-Schreibroutine
13. Prüfen auf Kanalfehler
14. Prüfen auf Statusfehler (RPDS)
15. Datenvergleich

Testmethodik:

entfällt; oben

6.5.6.26. Test 26: READ HEADER & DATA (MAINT. MODE)
DATA = 00
Kopf & Daten lesen (Wartungsmodus) Datenmuster 00(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	=	0
Spur	=	0
Sektor	=	0
Schlüsselnummer	=	0
Kopf CRC	=	C111(X)
Bytezähler	=	512
Datenmuster	=	00(X)
ECC	=	0

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register

5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler
12. Datenvergleich

Testmethodik:

entfällt

6.5.6.27. Test 27: READ HEADER & DATA (MAINT. MODE)
DATA = FF
Kopf & Daten lesen (Wartungsmodus) Datenmuster FF(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	=	0
Spur	=	0
Sektor	=	1
Schlüsselnummer	=	0
Kopf CRC	=	1010(X)
Bytezähler	=	512
Datenmuster	=	FF(X)
ECC	=	1B2FFF26(X)

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler
12. Datenvergleich

Testmethodik:

entfällt;

6.5.6.28. Test 28: READ HEADER & DATA (MAINT. MODE)
DATA = AA
Kopf & Daten lesen (Wartungsmodus) Datenmuster AA(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder = 0
Spur = 1
Sektor = 1
Schlüsselnummer = 0
Kopf CRC = 0101(X)
Bytezähler = 512
Datenmuster = AA(X)
ECC = D22AAC3B(X)

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler
12. Datenvergleich

Testmethodik:

entfällt;

6.5.6.29. Test 29: READ HEADER & DATA (MAINT. MODE)
DATA = 55
Kopf & Daten lesen (Wartungsmodus) Datenmuster 55(X)

Das Kommando wird mit folgenden Parametern ausgeführt:
Zylinder = 0
Spur = 1
Sektor = 1
Schlüsselnummer = 0
Kopf CRC = 0101(X)
Bytezähler = 512
Datenmuster = 55(X)
ECC = C905531D(X)

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler

12. Datenvergleich

Testmethodik:

entfällt;

6.5.6.30. Test 30: READ HEADER & DATA (MAINT. MODE)
FLOATING DATA 1
Kopf & Daten lesen (Wartungs-
modus) Datenmuster durchlaufende 1

Das Kommando wird mit folgenden Parametern ausgeführt:

```
  Zylinder        = 0
  Spur            = 1
  Sektor          = 1
  Schlüsselnummer = 0
  Kopf CRC        = 0101(X)
  Bytezähler      = 512
  Datenmuster     = durchlaufende 1
  ECC             = 51A24BA9(X)
```

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler
12. Datenvergleich

Testmethodik:

entfällt;

6.5.6.31. Test 31: READ HEADER & DATA (MAINT. MODE)
FLOATING DATA 0
Kopf & Daten lesen (Wartungs-
modus) Datenmuster durchlaufende 0

Das Kommando wird mit folgenden Parametern ausgeführt:

```
  Zylinder        = 0
  Spur            = 1
  Sektor          = 1
  Schlüsselnummer = 0
  Kopf CRC        = 0101
  Bytezähler      = 512
  Datenmuster     = durchlaufende 0
  ECC             = 4A8DB487(X)
```

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Kopf & Daten-Leseroutine
11. Prüfen auf Statusfehler
12. Datenvergleich

Testmethodik:

entfällt;

6.5.6.32. Test 32: WRITE DATA (MAINT. MODE)
DATA = FF00(X)
Datenschreiben im Wartungsmodus Datenmuster = FF00(X)

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	= 0
Spur	= 0
Sektor	= 0
Schlüsselnummer	= 0
Bytezähler	= 512
Datenmuster	= FF00(X)

Testschritte:

1. Einstellen Plattenadresse
2. Schreibpuffer füllen
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit (RPOF)
6. Setzen FMT22-Bit (RPOF)
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Daten-Schreibroutine
11. Prüfen auf Kanalfehler
12. Prüfen auf Statusfehler (RPDS)
13. Datenvergleich

Testmethodik:

entfällt;

```
6.5.6.33. Test 33: READ DATA (MAINT. MODE)
                   DATA = FF00(X)
                   Datenlesen im Wartungs-
                   modus Datenmuster = FF00(X)
```

Das Kommando wird mit folgenden Parametern ausgeführt:

```
  Zylinder         = 0
  Spur             = 0
  Sektor           = 1
  Schlüsselnummer  = 0
  Kopf CRC         = 1010(X)
  Bytezähler       = 512
  Datenmuster      = FF00(X)
  ECC              = 1B2FFF26(X)
```

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit im RPOF-Register
6. Setzen FMT22-Bit im RPOF-Register
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Daten-Schreibroutine
11. Prüfen auf Kommandofehler
12. Prüfen auf Statusfehler (RPDS)
13. Datenvergleich

Testmethodik:

entfällt;

```
6.5.6.34. Test 34: WRITE CHECK HEADER & DATA
                   (MAINT. MODE) DATA = 00
                   Kopf & Daten schreiben mit
                   Prüfung im Wartungsmodus
```

Das Kommando wird mit folgenden Parametern ausgeführt:

```
  Zylinder         = 0
  Spur             = 1
  Sektor           = 1
  Schlüsselnummer  = 0
  Kopf CRC         = D02D(X)
  Bytezähler       = 64
  Datenmuster      = 32 byte durchlaufende 1
                     32 byte durchlaufende 0
  ECC              = 2606DA72(X)
```

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. Einstellen Lesepuffer

4. MAP-Register des MBA (Datenkanal) einstellen
5. Einstellen RPCS1-Register
6. Setzen ECC INHIBIT-Bit im RPOF-Register
7. Setzen FMT22-Bit im RPOF-Register
8. Setzen Wartungsmodus
9. Setzen GO-Bit (RPCS1) Kommandoausführung
10. Ansprung SEARCH(Such)-Routine
11. Ansprung Daten-Schreibroutine
12. Prüfen auf Kommandofehler
13. Prüfen auf Statusfehler

Testmethodik:

entfällt;

6.5.6.35. Test 35: WRITE CHECK HEADER & DATA (MAINT. MODE) DATA = 00 Kopf & Daten schreiben mit Prüfung im Wartungsmodus

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder	= 0
Spur	= 1
Sektor	= 1
Schlüsselnummer	= 0
Kopf CRC	= D02D(X)
Bytezähler	= 80
Datenmuster	= 40 byte C7(X) 32 byte durchlaufende Muster
ECC	= xxxxx

Testschritte:

1. Einstellen Plattenadresse
2. Einstellen Simulations-Plattensektor
3. Einstellen Lesepuffer
4. MAP-Register des MBA (Datenkanal) einstellen
5. Einstellen RPCS1-Register
6. Setzen ECC INHIBIT-Bit im RPOF-Register
7. Setzen FMT22-Bit im RPOF-Register
8. Setzen Wartungsmodus
9. Setzen GO-Bit (RPCS1) Kommandoausführung
10. Ansprung SEARCH(Such)-Routine
11. Ansprung Daten-Schreibroutine
12. Prüfen auf Kommandofehler
13. Prüfen auf Statusfehler

Testmethodik:

entfällt;

6.5.6.36. Test 36: SECTOR SELECTION TEST
Sektorenauswahl

Das Kommando wird mit folgenden Parametern ausgeführt:

Zylinder = 0
Spur = 0
Sektor = 0-21
Schlüsselnummer = 0
Bytezähler = 512
Datenmuster = FF00(X)

Testschritte:

1. Einstellen Plattenadresse
2. Schreibpuffer füllen
3. MAP-Register des MBA (Datenkanal) einstellen
4. Einstellen RPCS1-Register
5. Setzen ECC INHIBIT-Bit (RPOF)
6. Setzen FMT22-Bit (RPOF)
7. Setzen Wartungsmodus
8. Setzen GO-Bit (RPCS1) Kommandoausführung
9. Ansprung SEARCH(Such)-Routine
10. Ansprung Daten-Schreibroutine
11. Prüfen auf Kanalfehler
12. Prüfen auf Statusfehler (RPDS)
13. Datenvergleich
14. Sektornummer aufzählen
15. Wiederholen, bis Sektornummer 22 erreicht

Testmethodik:

entfällt;

6.5.6.37. Test 37: ATTENTION ACTIVE BIT (ATA)
ATA-Bit testen

Dieser Test überprüft, daß zu jeder Zeit beim Setzen eines Fehlerbits in den Fehlerregistern auch das ATA-Bit gesetzt wird. Anschließend wird geprüft, ob durch das Schreiben einer 1 in das RPAS(ATTENTION SUMMARY)-Register das ATA-Bit gelöscht wird.

Testschritte:

1. Setzen eines Bits im Fehlerregister
2. Prüfen RPAS-Register
3. Prüfen ATA-Bit im RPCS1-Register
4. Löschen Fehlerbit
5. Löschen ATA-Bit durch Schreiben in RPAS-Register
6. Überprüfen, daß ATA-Bit gelöscht
7. Wiederholen für jedes Bit in allen 3 Fehlerregistern

Testmethodik:

Es wird vorausgesetzt, daß im Gerät Fehlerbedingungen nicht ständig bestehen, da sonst der Test nicht arbeiten kann.
Alle Fehlerbits sind auf der RG-BLP geodert. Es sind die entsprechenden

Auswahlsignale unter Beachtung des verwendeten Ports bei Fehlern zu überprüfen.

6.5.6.38. Test 38: FORMAT ERROR (FER) BIT TEST Formatfehler (FER-Bit)

Der Test überprüft die ordnungsgemäße Arbeit des Formatfehlerbits. Folgende Testparameter werden verwendet:

Zylinder = 0
Spur = 0
Sektor = 0
Schlüsselnummer = 0

Testschritte:

Subtest 1
1. Einstellen Simulationsplattensektor
2. Einstellen MAP-Register
3. Einstellen Plattenkontrollerregister zum Lesen
4. Setzen GO-Bit
5. Formatfehler erwarten

Subtest 2
1. Einstellen Simulationsplattensektor
2. Einstellen MAP-Register
3. Einstellen Plattenkontrollerregister zum Schreiben
4. Setzen GO-Bit
5. Formatfehler erwarten
6. Prüfen, daß keine Daten geschrieben wurden

Testmethodik:

entfällt;

6.5.6.39. Test 39: HEADER COMPARE ERROR (HCE) TEST Kopfvergleichsfehler (HCE)

Der Test überprüft, ob der Plattenkontroller Kopfvergleichsfehler ermittelt. Dazu wird versucht, mit einem Lese Kopf & Daten-Kommando, bzw. einem Schreibe Daten-Kommando auf eine inkorrekte Platten- bzw. Spuradresse zuzugreifen.

Testschritte:

Subtest 1
1. Einstellen Plattenkopf Zylinder 0, Spur 1, Sektor 1, Schlüsselnr. 1
2. Einstellen Kanal
3. Einstellen Plattenadresse Zylinder 1, Spur 1, Sektor 1, Schlüsselnr. 1
4. Einstellen Lese Kopf & Daten-Kommando
5. Setzen GO-Bit (Kommandoausführung)
6. Prüfen, daß HCE-Bit gesetzt
7. Prüfen, daß Ausnahmebedingung gesetzt
8. Prüfen, daß Kopf lesbar ist
9. Prüfen, daß Daten nicht lesbar sind

Subtest 2
1. Einstellen Plattenkopf Zylinder 0, Spur 1, Sektor 1, Schlüsselnr. 1
2. Einstellen Kanal
3. Einstellen Plattenadresse Zylinder 1, Spur 1, Sektor 1, Schlüsselnr. 1
4. Einstellen Schreibe Daten-Kommando
5. Setzen GO-Bit (Kommandoausführung)
6. Prüfen, daß HCE-Bit gesetzt
7. Prüfen, daß Ausnahmebedingung gesetzt
8. Prüfen, daß Kopf lesbar ist
9. Prüfen, daß Daten nicht geschrieben werden

Subtest 3
1. Einstellen Plattenkopf Zylinder 0, Spur 1, Sektor 1
2. Einstellen Kanal
3. Einstellen Plattenadresse Zylinder 0, Spur 0, Sektor 1
4. Einstellen Schreibe Daten-Kommando
5. Setzen GO-Bit (Kommandoausführung)
6. Prüfen, daß HCE-Bit gesetzt
7. Prüfen, daß Ausnahmebedingung gesetzt
8. Prüfen, daß Kopf lesbar ist
9. Prüfen, daß Daten nicht lesbar sind

Testmethodik:

Bei Fehler ist wahrscheinlich die SS-BLP die Ursache.

6.5.6.40. Test 40: HEADER CRC (HCRC) ERROR BIT TEST Kopf-CRC (HCRC)-Fehlerbit

Dieser Test überprüft, ob die Kopflogik des Plattenkontrollers ein inkorrektes Kopf-CRC-Wort finden kann. Dies wird für ein Schreib- bzw. Lese-Kommando ermittelt. Der Test setzt voraus, daß alle vorherigen Tests fehlerfrei gelaufen sind.

Testschritte:

Subtest 1
1. Initialisieren
2. Einstellen Plattenkopf
3. Einstellen Plattenadresse (RPDA, RPDC)
4. Einstellen Kanal MAP-Register
5. Kopf & Daten lesen
6. Prüfen, daß HCRC-Bit gesetzt
7. Prüfen, daß Ausnahmebedingung gesetzt
8. Prüfen, daß keine anderen Fehler vorliegen
9. Prüfen, daß Kopf beim Lesen keine Fehler bringt
10. Prüfen, daß keine Daten gelesen wurden

Subtest 2
1. Initialisieren
2. Einstellen Plattenkopf
3. Einstellen Plattenadresse (RPDA, RPDC)
4. Einstellen Kanal MAP-Register
5. Daten schreiben
6. Prüfen, daß HCRC-Bit gesetzt

7. Prüfen, daß Ausnahmebedingung gesetzt
8. Prüfen, daß keine anderen Fehler vorliegen
9. Prüfen, daß Kopf beim Lesen keine Fehler bringt
10. Prüfen, daß keine Daten geschrieben wurden

Testmethodik:

Die Kopfvergleichslogik befindet sich auf der SS-BLP.

6.5.6.41. Test 41: LAST SECTOR TRANSFERRED (LST) TEST Letzter Sektor erreicht

Dieser Test schreibt Daten nach Zylinder 814 oder 410 (abhängig vom Gerätetyp PP06/PP05), Spur 18 und Sektor 21.
Anschließend wird verglichen, daß das LST-Bit gesetzt ist, keine weiteren Fehlerbits auftreten und die Daten ordnungsgemäß übertragen werden. Folgende Testparameter finden Verwendung:

Zylinder = 410/814 PP05/PP06
Spur = 18
Sektor = 21
Bytezähler = 512
Datenmuster = FF00(x)

Testschritte:

1. Initialisieren
2. Einstellen Plattenkopf
3. Einstellen Plattenadresse
4. Einstellen Schreibe Daten-Kommando
5. FMT22-Bit setzen
6. GO-Bit setzen (Kommandoausführung)
7. Prüfen auf gesetztes LST-Bit
8. Prüfen auf andere Fehler
9. Datenvergleich
10. Wenn LST-Bit gesetzt, dieses durch Schreiben einer 0 ins RPDA-Register rücksetzen
11. Prüfen, ob Löschanweisung erfolgreich

Testmethodik:

entfällt;

6.5.6.42. Test 42: ADDRESS OVERFLOW ERROR (ADE) BIT TEST Adressenüberlauffehler ADE-Bit

Dieser Test schreibt Daten nach Zylinder 814 oder 410 (abhängig vom Gerätetyp PP06/PP05), Spur 18, Sektor 21 und einem Bytezähler von 513 byte.
Damit wird geprüft, daß das ADE-Bit gesetzt wird, nachdem mehr als 512 byte übertragen wurden. Folgende Testparameter finden Verwendung:

Zylinder = 410/814 PP05/PP06
Spur = 18

Sektor = 21
Bytezähler = 513
Datenmuster = FF00(X)

Testschritte:

1. Initialisieren
2. Einstellen Plattenkopf
3. Einstellen Plattenadresse
4. Schreibpuffer füllen
5. Einstellen MAP-Register
6. Einstellen Schreibe Daten-Kommando
7. Setzen GO-Bit (RPCS1) Kommandoausführung
8. Prüfen, daß ADE-Bit gesetzt
9. Prüfen, daß LST-Bit gesetzt
10. Prüfen, daß EXCEPTION-Bit gesetzt (Ausnahmebedingung)
11. Prüfen, daß keine anderen Fehler aufgetreten
12. Datenvergleich (512 byte müssen richtig übertragen sein)

Testmethodik:

Die Logik zur Beschaltung von ADE- und LST-Bits ist auf der SS-BLP enthalten.

6.5.6.43. Test 43: REGISTER MODIFICATION REFUSED Registermodifizierung verhindern

Im Wartungsmodus wird ein Schreibe Daten-Kommando angewiesen. Während der Datentransfer läuft wird versucht, in jedes der folgenden Register zu schreiben:
RPCS1, RPDA, RPDCA, RPOF, RPER1, RPER2, RPER3
Anschließend wird geprüft, ob das RMR-Bit (REGISTER MODIFICATION REFUSED) gesetzt ist. Nunmehr wird der Datentransfer beendet. Dabei kommen folgende Testparameter zur Anwendung:
Zylinder = 0
Spur = 0
Sektor = 0
Bytezähler = 0
Datenmuster = FF00(X)

Testschritte:

1. Initialisieren
2. Einstellen Simulationsplatte
3. Einstellen Plattenadresse
4. Schreibpuffer füllen
5. Einstellen MAP-Register des Kanals
6. Einstellen Schreibe Daten-Kommando
7. Setzen GO-Bit (Kommandoausführung)
8. Schreibversuch in ein Register
9. Prüfen, ob RMR-Bit gesetzt
10. Fortsetzen Datentransfer
11. Prüfen auf andere Fehler
12. Datenvergleich (zum aktuellen Zeitpunkt)
13. Wiederholen für jedes oben aufgeführte Register

Testmethodik:

siehe Testbescheibung

6.5.6.44. Test 44: INVALID ADDRESS ERROR (IAE) Fehler ungültige Adresse

Mit einem Kopf & Daten-Lese-Kommando wird versucht, nacheinander auf eine ungültige Zylinder-, Sektor- und Spuradresse zuzugreifen. Für die nachfolgenden 3 Subtests kommmen folgende Testparameter zur Anwendung:

	Subtest 1	Subtest 2	Subtest 3
Zylinder	411/815 (PP05/PP06)	0	0
Spur	0	0	20
Sektor	0	22 (16-bit-Modus)	0

Testschritte:

1. Initialisieren
2. Einstellen Plattenadresse
3. Einstellen 16-bit-Modus
4. Einstellen Kopf & Daten Lese-Kommando
5. Setzen GO-Bit (Kommandoausführung)
6. Prüfen, daß IAE-Bit gesetzt
7. Prüfen, daß EXCEPTION-Bit gesetzt

Testmethodik:

entfällt;

6.5.6.45. Test 45: ILLEGAL FUNCTION (ILF) BIT TEST Illegale Funktion ILF-Bit

Dieser Test überprüft, ob alle illegalen Funktionscodes den Fehler "Illegale Funktion" erzeugen. Es folgen die möglichen illegalen Funktionscodes (hexadezimal):
OA, OB, OD, OE, OF, 10, 11, 12, 13, 16, 17, 1A, 1B, 1E, 1F

Testschritte:

1. Initialisieren
2. Einstellen Funktionscode
3. Einstellen Wartungsmodus
4. Setzen GO-Bit (Kommandoausführung)
5. Prüfen, daß ILF-Bit gesetzt
6. Wiederholen für alle obigen Codes

Testmethodik:

Prüfen der Funktionsentschlüsselung auf der RG-BLP.

6.5.6.46. Test 46: ILLEGAL REGISTER (ILR) BIT TEST Illegales Register ILR-Bit

Dieser Test versucht, auf ein illegales Register zuzugreifen. Es wird geprüft, ob der Fehler "Illegales Register" entschlüsselt wird.

Testschritte:

1. Initialisieren
2. Schreibversuch auf ein illegales Register
3. Prüfen, ob ILR-Bit gesetzt

Testmethodik:

Es sind die entsprechenden Setz- und Löschsignale für das ILR-Flip-Flop zu überprüfen.

6.5.6.47. Test 47: DEVICE TIMING ERROR (DTE) TEST Zeitablauffehler des Gerätes

Es wurde ein Kopf & Daten Lese-Kommando gestartet. Nachdem der Kopfteil korrekt gelesen wurde, wird (im Wartungsregister) kein Synchron-Byte entschlüsselt. Dann werden die normalen Daten- und Sektorimpulse für 24 byte erzeugt. Danach folgen 536 Sektorimpulse ohne Datenimpuls. Wenn der letzte Sektorimpuls kommt, müßte das DTE-Bit gesetzt werden. Folgende Parameter werden angewendet: Zylinder 0, Spur 0, Sektor 0, Bytezähler 520

Testschritte:

1. Initialisieren
2. Einstellen Lesepuffer
3. Einstellen Simulationsplatte
4. Einstellen MAP-Register des MBA
5. Einstellen Plattenkontrollerregister
6. Setzen GO-Bit (Kommandoausführung)
7. Index Sektorimpuls erzeugen
8. Lesen PREHEADER 9 byte
9. Lesen Synchron-Byte 19(X)
10. Lesen HEADER 10 byte
11. Lesen PREDATA AREA 11 byte
12. Lesen 24 Normal-Byte ohne Synchron-Byte
13. 536 Sektorimpulse ohne Datenimpulse
14. Prüfen, ob DTE-Bit gesetzt

Testmethodik:

Der Zeitablauffehler (DTE) entsteht als Resultat einer laufenden Lese- oder Schreibfunktion, wenn Sektorimpulse entstehen, welche normalerweise nicht zu verwenden sind.
Die Logik, die zur Zeitfehlererkennung führt, ist im SN-BLP zu finden.

6.5.7. Auflistung der Tests

Test 1: SUBSYSTEM REGISTER ADDRESS TEST
Subsystem Registeradressierung
Test 2: QUALIFICATION TEST
Qualifikation
Test 3: RPCS1 REGISTER
RPCS1 (Statuskommando)-Register
Test 4: RPDA - REGISTER
RPDA (Diskadreß) - Register
Test 5: RPER1 - REGISTER
RPER1 (Fehler) - Register
Test 6: RPMR REGISTER
RPMR (Wartungs) - Register
Test 7: RPOF - REGISTER
RPOF (Versatz) - Register
Test 8: RPDC - REGISTER
RPDC (Soll-Zylinderadreß)-Register
Test 9: RPER2 - REGISTER
RPER2 (Fehler 2)- Register
Test 10: RPER3 REGISTER
RPER3 (Fehler 3)-Register
Test 11: MBA INIT
MBA Initialisierung
Test 12: DRIVE CLEAR
Gerät rücksetzen
Test 13: PACK ACKNOWLEDGE AND READ IN PRESET COMMAND
Erkennungskommando, Anfangsrückstellen
Test 14: NOOP COMMAND
Keine Operation
Test 15: SEEK COMMAND
Positionierkommando
Test 16: RECALIBRATE COMMAND
Nullpositionierung
Test 17: OFFSET COMMAND
Versatzkommando
Test 18: RETURN TO CENTERLINE
Zurück zur Ausgangsposition
Test 19: SEARCH COMMAND & LOOK AHEAD REGISTER
Suchkommando und Vorschauregister
Test 20: WRITE HEADER & DATA (MAINT. MODE) DATA = 00
Kopf & Daten schreiben (Wartungsmodus) Datenmuster 00(X)
Test 21: WRITE HEADER & DATA (MAINT. MODE) DATA = FF
Kopf & Daten schreiben (Wartungsmodus) Datenmuster FF(X)
Test 22: WRITE HEADER & DATA (MAINT. MODE) DATA = AA
Kopf & Daten schreiben (Wartungsmodus) Datenmuster FF(X)
Test 23: WRITE HEADER & DATA (MAINT. MODE) DATA = 55
Kopf & Daten schreiben (Wartungsmodus) Datenmuster 55(X)
Test 24: WRITE HEADER & DATA (MAINT. MODE) FLOATING DATA 1
Kopf & Daten schreiben (Wartungsmodus) Datenmuster durchlaufende 1
Test 25: WRITE HEADER & DATA (MAINT. MODE) FLOATING DATA 0
Kopf & Daten schreiben (Wartungsmodus) Datenmuster durchlaufende 0
Test 26: READ HEADER & DATA (MAINT. MODE) DATA = 00
Kopf & Daten lesen (Wartungsmodus) Datenmuster 00(X)
Test 27: READ HEADER & DATA (MAINT. MODE) DATA = FF
Kopf & Daten lesen (Wartungsmodus) Datenmuster FF(X)
Test 28: READ HEADER & DATA (MAINT. MODE) DATA = AA
Kopf & Daten lesen (Wartungsmodus) Datenmuster AA(X)

Test 29: READ HEADER & DATA (MAINT. MODE) DATA = 55
Kopf & Daten lesen (Wartungsmodus) Datenmuster 55(X)
Test 30: READ HEADER & DATA (MAINT. MODE) FLOATING DATA 1
Kopf & Daten lesen (Wartungsmodus) Datenmuster durchlaufende 1
Test 31: READ HEADER & DATA (MAINT. MODE) FLOATING DATA 0
Kopf & Daten lesen (Wartungsmodus) Datenmuster durchlaufende 0
Test 32: WRITE DATA (MAINT. MODE) DATA = FF00(X)
Datenschreiben im Wartungsmodus Datenmuster = FF00(X)
Test 33: READ DATA (MAINT. MODE) DATA = FF00(X)
Datenlesen im Wartungsmodus Datenmuster = FF00(X)
Test 34: WRITE CHECK HEADER & DATA (MAINT. MODE) DATA = 00
Kopf und Daten schreiben mit Prüfung im Wartungsmodus
Test 35: WRITE CHECK HEADER & DATA (MAINT. MODE) DATA = 00
Kopf & Daten schreiben mit Prüfung im Wartungsmodus
Test 36: SECTOR SELECTION TEST
Sektorenauswahl
Test 37: ATTENTION ACTIVE BIT (ATA)
ATA-Bit testen
Test 38: FORMAT ERROR (FER) BIT TEST
Formatfehler (FER-Bit)
Test 39: HEADER COMPARE ERROR (HCE) TEST
Kopfvergleichsfehler (HCE)
Test 40: HEADER CRC (HCRC) ERROR BIT TEST
Kopf-CRC (HCRC)-Fehlerbit
Test 41: LAST SECTOR TRANSFERRED (LST) TEST
Letzter Sektor erreicht
Test 42: ADDRESS OVERFLOW ERROR (ADE) BIT TEST
Adressenüberlauffehler ADE-Bit
Test 43: REGISTER MODIFICATION REFUSED
Registermodifizierung verhindern
Test 44: INVALID ADDRESS ERROR (IAE)
Fehler ungültige Adresse
Test 45: ILLEGAL FUNCTION (ILF) BIT TEST
Illegale Funktion ILF-Bit
Test 46: ILLEGAL REGISTER (ILR) BIT TEST
Illegales Register ILR-Bit
Test 47: DEVICE TIMING ERROR (DTE) TEST
Zeitablauffehler des Gerätes

Test 29: READ HEADER & DATA (MAINT. MODE) DATA = 55
Kopf & Daten lesen (Wartungsmodus) Datenmuster 55(X)
Test 30: READ HEADER & DATA (MAINT. MODE) FLOATING DATA 1
Kopf & Daten lesen (Wartungsmodus) Datenmuster durchlaufende 1
Test 31: READ HEADER & DATA (MAINT. MODE) FLOATING DATA 0
Kopf & Daten lesen (Wartungsmodus) Datenmuster durchlaufende 0
Test 32: WRITE DATA (MAINT. MODE) DATA = FF00(X)
Datenschreiben im Wartungsmodus Datenmuster = FF00(X)
Test 33: READ DATA (MAINT. MODE) DATA = FF00(X)
Datenlesen im Wartungsmodus Datenmuster = FF00(X)
Test 34: WRITE CHECK HEADER & DATA (MAINT. MODE) DATA = 00
Kopf und Daten schreiben mit Prüfung im Wartungsmodus
Test 35: WRITE CHECK HEADER & DATA (MAINT. MODE) DATA = 00
Kopf & Daten schreiben mit Prüfung im Wartungsmodus
Test 36: SECTOR SELECTION TEST
Sektorenauswahl
Test 37: ATTENTION ACTIVE BIT (ATA)
ATA-Bit testen
Test 38: FORMAT ERROR (FER) BIT TEST
Formatfehler (FER-Bit)
Test 39: HEADER COMPARE ERROR (HCE) TEST
Kopfvergleichsfehler (HCE)
Test 40: HEADER CRC (HCRC) ERROR BIT TEST
Kopf-CRC (HCRC)-Fehlerbit
Test 41: LAST SECTOR TRANSFERRED (LST) TEST
Letzter Sektor erreicht
Test 42: ADDRESS OVERFLOW ERROR (ADE) BIT TEST
Adressenüberlauffehler ADE-Bit
Test 43: REGISTER MODIFICATION REFUSED
Registermodifizierung verhindern
Test 44: INVALID ADDRESS ERROR (IAE)
Fehler ungültige Adresse
Test 45: ILLEGAL FUNCTION (ILF) BIT TEST
Illegale Funktion ILF-Bit
Test 46: ILLEGAL REGISTER (ILR) BIT TEST
Illegales Register ILR-Bit
Test 47: DEVICE TIMING ERROR (DTE) TEST
Zeitablauffehler des Gerätes

6.6. PVRDA: PM80-Test ohne Platte

6.6.1. Zusammenfassung

Das Diagnoseprogramm PVRDA (DOK.-NR. 1.57.700606.4) ist ein Programm der Ebene 3 (Stand-alone). Es besteht aus einer Testserie, welche die Funktionsfähigkeit des PM80 Plattensubsystem ausschließlich und unabhängig vom Speichergerät prüft. Das Programm dient der Ermittlung von Fehlern im PM80-Adapter.

6.6.2. Testbedingungen

6.6.2.1. Hardware

- RVS A 40 (mit Minimalkonfiguration)
- MSBUS-Adapter K 2816
- Konsolterminal
- RHA40 mit ein oder mehreren PM80
- Testeinheit, bestehend aus ein bis acht PM80-Adapter

6.6.2.2. Software

Das Programm PVRDA ist nur in Verbindung mit dem Diagnose-Supervisor (DS) lauffähig. Beide, das Programm PVRDA und der DS, müssen vor dem Start geladen werden.

6.6.3. Voraussetzungen

Für eine Abarbeitung des Programms wird die volle Funktionstüchtigkeit aller Elemente der CPU vorausgesetzt.Der DS muß geladen sein.

6.6.4. Bedienungshinweise

Die Bedienungsanleitung des DS ist für den Lade-und Startvorgang mitzuverwenden. Der DS meldet sich mit:

```
DS>
```

Das Testprogramm PVRDA ist zu laden:

```
DS> LOAD PVRDA.EXE
```

Vor dem Programmstart muß das Testgerät angeschlossen werden (ATTACH) und für den Test logisch ausgewählt werden (SELECT).

```
DS> ATTACH RHA40
DEVICE LINK? HUB
DEVICE NAME? RHn
TR? number? (dezimal 1-15) Standard <8>
BR? level? (dezimal 4-7)   Standard <5>
DS> SELECT RHn
DS> ATTACH PM80
DEVICE LINK? RHn
DEVICE NAME? DRAn
DS> SELECT DRAn
```

Die Übertragungsstufe (TR) und die Busanforderungsstufe (BR) sind zusätzliche Informationen für eine korrekte Geräteeingliederung.

Programmstart:

```
DS> START
```

Weitere Bedienungshinweise sind der DS-Beschreibung zu entnehmen.

6.6.5. Funktionsbeschreibung

6.6.5.1. Übersicht

Das Programm beinhaltet 80 Tests. Eine Auflistung aller Tests ist im Abschnitt 6.6.8. zu finden.
Das Programm wird in fünf Objektmodule unterteilt:

Modul 1:	enthält globale Daten und Subroutinen
Modul 2:	Test 1-18
Modul 3:	Test 19-35
Modul 4:	Test 36-75
Modul 5:	Test 76-80

6.6.5.2. Programmgröße

Der Speicherbedarf für dieses Programm beträgt 53760. byte,105 Pages.

6.6.5.3. Programmlaufzeit

Die Programmlaufzeit für einen fehlerfreien Standarddurchlauf beträgt ca. 7,5s.

6.6.5.4. Abarbeitungsbedingungen

entfällt

6.6.5.5. Flagverwendung

entfällt

6.6.5.6. Allgemeine Programmanforderungen

entfällt

6.6.5.7. Fehlerermittlung

Das Programm PVRDA liefert umfangreiche Fehlermeldungen mit der Absicht,die Fehlersuche zu unterstützen.
Die erste Zeile der Fehlermeldung enthält den Durchlaufzähler, die Testnummer und die Fehlernummer. Dieser Zeile folgt die eigentliche Fehlermitteilung. Es können mehrere Zeilen sein,welche eine umfangreiche Beschreibung des Fehlers und eine Auflistung von fehlerhaften Moduln in der Reihenfolge der abnehmenden Wahrscheinlichkeit geben. Wenn notwendig werden Soll-und Istdaten ausgeschrieben.
An dem nebenstehenden Beispiel soll eine Fehlermitteilung erläutert werden.

```
(1)  ******** PVRDA PM03/PM05/PM80 DISKLESS DIAGNOSTIC - 1.0 ********
(2)   Pass 1, test 20, subtest 0, error 2, 23-JUL-1987 11:19:39.82
(3)   Hard error while testing dra0:  CANNOT WRITE ZEROS IN RMDC

(4)  GOOD:    FFFF(X)
(5)  BAD:     0000(X)
(6)  XOR:     FFFF(X)

(7)  BAD ADDRESS:      60010428(X)
(8)  FAILING-MODULE(S)  -  IF,DS
(9)  ********  End of Hard error number 2 ********
```

Erläuterung:
(1) Überschrift mit Testname und Versionsnummer
(2) Durchlaufzähler Testnummer mit Subtest-und Errornummer,Startzeit
(3) Fehlerart
(4) Sollwert
(5) Istwert
(6) Differenz (XOR Vergleich)
(7) fehlerhaftes Register (Adresse)
(8) fehlerhafte Module
(9) Beendigungsmitteilung

6.6.5.8. Programmitteilungen

Eine Durchlaufendemeldung wird für alle Adapter in der Testschlange ausgegeben. Sie enthält eine Mitteilung und die Fehlersumme.

6.6.5.9. Leistungsparameter des Testobjektes

entfällt

6.6.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.6.5.11. Testfolge bei mehreren Einheiten

entfällt

6.6.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.6.5.13. Testsektionen

Das Programm PVRDA enthält eine Testsektion.

DEFAULT: Abarbeitung aller Tests

6.6.6. Testbeschreibungen

6.6.6.1. Test 1: PROBE DEVICE REGISTERS
Geräteregister ansprechen

In diesem Test wird geprüft, ob alle Geräteregister lesbar sind. Dazu wird der MBA initialisiert. Bei fehlerhaftem Ablauf erfolgt Fehler 1. Anschließend wird der MBA-Status gelöscht. Läßt sich das nicht fehlerfrei realisieren, kommt es zum Fehler 2. Sind die Geräteregister nicht lesbar, kommt es zum Fehler 3.

6.6.6.2. Test 2: QUALIFICATION TESTS
Qualifikationsprüfung

Es wird getestet, ob der MSBUS aktiv und ob die Testeinheit vorhanden und verfügbar ist. Dazu muß das Bit DVA (Gerät verfügbar) im Steuerregister RMCS1 und das Bit DRY (Gerät fertig) im Gerätestatusregister (RMDS) gesetzt sein.
Zum Fehler 1 kommt es,wenn ein CONTROL BUS PARITY-ERROR festgestellt wird.Wenn ein nicht vorhandenes Gerät erkannt wurde,wird Fehler 2 angegeben.

6.6.6.3. Test 3: CTOD TEST
Kontroller zum Gerät - Test

Es wird geprüft, ob ein Datentransport zum und vom Speicher erfolgen kann, und im einzelnen, ob die Steuerleitung CTOD funktioniert.
Der Test schreibt dazu Einsen in die REMOTE-Register.Anschließend werden die Register zweimal ausgelesen. Es wird kontrolliert,ob Einsen zurückgelesen werden können. Sonst kommt es zum Fehler 1.

6.6.6.4. Test 4: MSBUS INITIALIZE TEST
MSBUS Initialisierungstest

Das Kontroller-Löschen wird benutzt, um die ausgewählte Einheit zu initialisieren. Anschließend werden die MSBUS-Adapter-Register gelesen und kontrolliert, ob mindestens ein Bit gelöscht ist (Fehler 1). Die MSBUS-Apapter-Register sind voreingestellt auf einen von Null verschiedenen Wert vor Kontroller-Löschen.

6.6.6.5. Test 5: CLEAR STUCK ACTIVE TEST
Löschen ständig aktiv

Es wird geprüft, ob MBA CLR L nicht ständig im aktiven Zustand ist. Das Kontroller-Löschen wird benutzt, um die ausgewählte Einheit zu initialisieren. Danach werden Einsen in das Fehlerregister 1 und 2 und ins Wartungsregister 1 geschrieben. Wenn irgendeines der Bits rückgelesen werden kann, ist der Test in Ordnung. Ansonsten kommt es zum Fehler 1.

6.6.6.6. Test 6: TRISTATE TRANSFER TEST
Tristate - Übertragungstest

Der Test prüft, ob der Weg zum und vom MSBUS-Adapter TRISTATE-Bus nicht fest auf Eins oder Null steht, und das jede Bitposition unabhängig voneinander ist.
Dazu werden die MSBUS-Adapter-Register mit einem von Null verschiedenen Wert geladen. Anschließend werden diese ausgewählten Register durch eine MOVE-Instruktion gelöscht. Es wird dann kontrolliert, ob die Bitpositionen Null sind. Ansonsten kommt es zum Fehler 1. Das Programm löscht die MSBUS- Adapter-Register. Danach werden sie mit Einsen beschrieben und ausgelesen. Sind danach nicht alle Bitpositionen 1, kommt es zum Fehler 2. Am Schluß des Testes wird ein Bitmuster in die Register geschrieben und zurückgelesen. Bei Nichtübereinstimmung, kommt es zum Fehler 3. Der Abschlußtest wird für alle Bitpositionen wiederholt.

6.6.6.7. Test 7: REGISTER SELECT TEST
Registerauswahltest

Der Test prüft die Registerauswahlleitungen. Jede Auswahlleitung wird getestet durch Schreiben von Nullen in jene Geräteregister, für welche die Leitung Null sein muß. Anschließend werden Einsen geschrieben in die Register, für welche die Leitung Eins sein muß.
Es werden die Registerauswahlleitungen 1,2,4 und 8 getestet. Die Auswahlleitung 16 wird im Test 21 geprüft.

6.6.6.8. Test 8: DRIVE TYPE TEST
Gerätetyp - Test

Der Test liest das Register RMDT aus und prüft, ob es übereinstimmt mit einem Einkanal-oder Zweikanal PM80 (Fehler 1).

6.6.6.9. Test 9: DEVICE AVAILABLE TEST
Gerät verfügbar - Test

Bei diesem Test wird geprüft, ob das Bit DVA (Gerät verfügbar) im Register RMCS1 gesetzt ist.Wenn DVA nicht gesetzt ist erfolgt Fehler 1.

6.6.6.10. Test 10: HOLDING REGISTER TRANSFER TEST
Halteregister (RMHR) - Übertragungstest

Dieser Test prüft, ob das HOLDING-Register nicht fest auf Eins oder Null steht und ob sich die Bits nicht gegenseitig beeinflussen.
Dazu transportiert das Programm zuerst Einsen dann Nullen in das Reg. RMHR und

prüft,ob keins der Bits auf Eins steht (Fehler 1).
Anschließend transportiert das Programm zuerst Nullen dann Einsen in das Reg. RMHR und prüft,ob keins der Bits danach auf Null steht (Fehler 2). Zum Schluß des Tests wird ein Bitmuster durch das Register geschoben und geprüft, ob sich die Bits nicht gegenseitig beeinflussen (Fehler 3).

6.6.6.11. Test 11: CONTROL STATUS #1 TRANSFER TEST Steuerstatusregister 1 (RMCS1) - Test

Der Test prüft, ob die Bits 01 bis 05 des Reg. RMCS1 nicht fest auf Null oder Eins stehen und ob sie sich nicht gegenseitig beeinflussen.
Zuerst werden dazu Einsen in das Register RMCS1 (Bits 01-05) geschrieben und anschließend gelöscht. Konnte das Register nicht gelöscht werden, kommt es zum Fehler 1. Danach schreibt der Test Einsen in das Reg. RMCS1 und überschreibt es mit Nullen. Das GO-Bit wird nicht mit ausgetestet.Zum Fehler 2 kommt es, wenn keine Nullen geschrieben werden konnten.
Anschließend wird das Reg. RMCS1 gelöscht. Der Test schreibt dann Einsen von Bitposition 01 bis 05 und prüft, ob die Bits danach auch nicht auf Null stehen (Fehler 3). Das GO-Bit ist hier ebenfalls aus dem Test genommen.Am Ende des Tests wird ein Bitmuster durch das Reg. RMCS1 geschoben und kontrolliert, ob die angrenzenden Bits unabhängig voneinander sind (Fehler 4).

6.6.6.12. Test 12: ERROR REGISTER 1 TRANSFER TEST Fehlerregister 1 (RMER1) - Übertragungstest

Der Test prüft, ob das Register RMER1 nicht auf Eins oder Null festgeklemmt ist und daß sich die Bits nicht gegenseitig beeinflussen. Dabei wird jeder einzelne Test in drei Abschnitten ausgeführt,wobei jeder Teil die Bits testet,deren vorgegebene Funktionen vom selben Modul abgeleitet sind.Der Test erfolgt deshalb in drei Abschnitten, um die wahrscheinlichen Fehler auf ein oder zwei Module zu begrenzen.
Als erstes werden Einsen in das Register RMER1 geschrieben und anschließend in drei Abschnitten verschiedene Bits des Registers gelöscht. Konnten diese nicht gelöscht werden, kommt es entweder zum Fehler 1,Fehler 2 oder Fehler 3.
Als nächstes werden Einsen in das Register RMER1 geschrieben und anschließend mit Nullen überschrieben. Dieser Test wird ebenfalls in drei Abschnitten ausgeführt. Konnten die Bits nicht gelöscht werden, kommt es entweder zum Fehler 4,Fehler 5 oder Fehler 6.
Im dritten Teil werden zuerst Nullen und dann Einsen in das Register RMER1 geschrieben. Wurde das Register nicht beschrieben, kommt es zum Fehler 7.
Am Ende des Testes wird ein Bitmuster durch das Register RMER1 geschoben und kontrolliert,ob sich die Bits nicht beeinflussen (Fehler 8).

6.6.6.13. Test 13: ERROR REGISTER 2 TRANSFER TEST Fehlerregister 2 (RMER2) - Übertragungstest

Der Test prüft, ob das Register RMER2 nicht auf Eins oder Null festgeklemmt ist und daß sich die Bits nicht gegenseitig beeinflussen.
Dazu schreibt der Test als erstes Einsen in das Register RMER2 und löscht es dann. Die Prüfung erfolgt danach in drei Abschnitten.
Im ersten Abschnitt werden alle unbenutzten Bits des Registers auf Null getestet. Sind diese danach nicht Null, erfolgt Fehler 1.

Im zweiten Abschnitt werden die Bits OPE,IVC und LSC auf Null getestet. Konnten sie nicht gelöscht werden, erfolgt Fehler 2.
Im dritten Abschnitt wird das Bit DPE auf Null getestet. Ansonsten erfolgt Fehler 3. Der Test erfolgt deshalb in drei Abschnitten, um die wahrscheinlichen Fehler auf ein oder zwei Module zu begrenzen.
Anschließend schreibt der Test Nullen und dann Einsen in das Fehlerregister 2. Die Prüfung erfolgt danach in zwei Abschnitten.
Als erstes werden die Bits OPE,IVC und LSC auf Null getestet. Sind diese nicht Null, erfolgt Fehler 4. Wurden die Bits LBC und DPE nicht gelöscht, kommt es zum Fehler 5.
Als nächstes schreibt der Test zuerst Nullen und dann Einsen in das Fehlerregister 2. Es wird dann geprüft, ob danach alle unbelegten Bits gelöscht (Fehler 6) und alle belegten gesetzt sind (Fehler 7).
Am Schluß dieses Testes wird ein Bitmuster durch das Register geschoben und geprüft,ob die Bitpositionen unabhängig sind. Ansonsten erfolgt Fehler 8.

6.6.6.14. Test 14: CLEAR OFFSET STUCK ACTIVE TEST
Offsetlöschen ständig aktiv

Es wird geprüft, ob das Signal,welches den Offsetmodus löscht, nicht dauernd im aktiven Zustand ist. Dazu schreibt der Test eine Eins in das Offset-Direction-Bit,welches durch das Signal gelöscht wurde und prüft, ob eine Eins zurückgelesen werden kann. Ansonsten erfolgt Fehler 1.

6.6.6.15. Test 15: OFFSET REGISTER TRANSFER TEST
Offsetregister (RMOF) - Übertragungstest

Der Test prüft, ob das Offsetregister nicht fest auf Eins oder Null steht, und ob sich die angrenzenden Bits nicht gegenseitig beeinflussen.
Das Offsetregister wird dazu mit Einsen beschrieben und dann mit Nullen. Es wird danach zweimal gelesen. Dabei wird kontrolliert, ob alle unbelegten Bits Null sind (Fehler1), und ob alle belegten Bits ebenfalls gelöscht sind (Fehler 2).
Das Offsetregister wird anschließend mit Nullen und danach mit Einsen beschrieben. Es wird kontrolliert, ob alle unbelegten Bits =0 (sonst Fehler 3) und alle belegten Bits =1 sind (sonst Fehler 4).
Am Ende dieses Testes wird ein Bitmuster durch das Register geschoben und geprüft, ob die Bits unabhängig voneinander sind (sonst Fehler 5).

6.6.6.16. Test 16: SERIAL NUMBER TEST
Seriennummernregister (RMSN) - Test

Der Test prüft, ob sich das Register RMSN lesen läßt. Dazu wird das Register RMSN mehrere Male gelesen und kontrolliert, ob die Speichernummer jedesmal die gleiche ist (Fehler 1).

6.6.6.17. Test 17: CONTROL BUS PARITY GENERATION
Prüfzeichenlogiktest

In diesem Test wird die Prüfzeichenlogik für den MSBUS-Kontroller geprüft.
Der Test verschiebt dazu ein Bitmuster durch das Plattenadreßregister.
Danach wird jedes Muster zurückgelesen und das Bit MCPE (Paritätsfehler) muß 0 sein.

Die folgenden Testmuster können anstelle des verschieblichen Musters benutzt werden.

Datenmuster	MCPE
000000	0
056747	0
135672	0
163135	0

6.6.6.18. Test 18: RMDA,RMDC FAULT TEST RMDA,RMDC Fehlertest

Der Test prüft, ob Fehlerbedingungen vorliegen, welche das Beschreiben der Register RMDC und RMDA verhindern.
Dazu beschreibt und liest der Test sowohl das Register RMDC und RMDA mit Nullen und anschließend Einsen. Der Test ist beendet, wenn jedes Register mit Einsen beschrieben werden konnte (Fehler 1).

6.6.6.19. Test 19: DISK ADDRESS TRANSFER TEST Plattenadressen - Übertragungstest

Der Test prüft, ob das Register RMDA nicht fest auf Null oder Eins steht und daß kein Bit gestört ist.
Dazu wird eine Geräteadresse mit dem Wert ungleich 0 vorgegeben.
Anschließend wird das Register gelöscht. Der Test liest dann das Register RMDA und prüft, ob keins der Bits auf Eins steht (Fehler 1). Danach wird das Register RMDA mit Einsen geladen und es wird geprüft, daß keins der Bits danach auf Null steht (Fehler 1).
Zum Schluß des Testes wird ein Bitmuster durch das Register RMDA geschoben und es wird geprüft ,ob sich die Bits nicht gegenseitig beeinflussen (Fehler 3).

6.6.6.20. Test 20: DESIRED CYLINDER TRANSFER TEST Zylinderadressen - Übertragungstest

Der Test prüft, ob das Zylinderadreßregister RMDC sich nicht auf Null oder Eins festgeklemmt hat, und daß sich kein Bit untereinander stört.
Als erstes wird dazu das Register RMDC mit Einsen geladen und kontrolliert,ob danach alle unbelegten Bits des Registers =0 sind (Fehler 1).
Als nächstes schreibt der Test zuerst Einsen, dann Nullen in das Register RMDC und prüft, ob das Register danach nicht auf Eins steht (Fehler 2).
Anschließend werden zuerst Nullen und dann Einsen in das Register RMDC geschrieben .Es wird dann geprüft, ob alle Bits auf Eins stehen (Fehler 3).
Am Ende des Testes wird ein Bitmuster durch das Register RMDC geschoben und kontrolliert,ob die Bits unabhängig voneinander sind (Fehler 4).

6.6.6.21. Test 21: ILEGAL REGISTER TEST Test auf illegales Register

Der Test dient zur Erkennung des illegalen Registerfehlers (ILR).
Als erstes wird das Register RMER1 gelöscht und kontrolliert, ob das Bit ILR

gelöscht ist (Fehler1).
Anschließend werden alle legalen Register beschrieben und geprüft, ob das Bit ILR nicht gesetzt ist. Sonst kommt es zum Fehler 2.
Als letztes wird versucht ein illegales Register zu beschreiben. Es wird dann geprüft, ob das ILR-Bit gesetzt ist. Wurde es nicht gesetzt, kommt es zum Fehler 3.

6.6.6.22. Test 22: RESET GO BY INIT TEST Rücksetzen GO durch INIT

Der Test setzt das GO. Es wird geprüft, ob durch INIT das GO gelöscht wird. Konnte GO nicht durch INIT gelöscht werden, kommt es zum Fehler 1.

6.6.6.23. Test 23: DIAGNOSTIC MODE TEST Diagnosemodus - Test

Der Test prüft, ob der Diagnosemodus (DMD), nicht fest auf Eins oder Null steht. Dazu wird das Bit DMD im Register RMMR1 nach Initialisierung auf Null kontrolliert.Ist es nicht gelöscht, erfolgt Fehler 1.
Anschließend wird das Bit DMD gesetzt und wieder gelöscht. Ist es danach nicht gelöscht, erfolgt Fehler 2.
Als nächstes wird DMD gesetzt und kontrolliert. Wurde es nicht gesetzt, kommt es zum Fehler 3.

6.6.6.24. Test 24: MEDIUM ONLINE - TEST Gerät online - Test

Der Test prüft, ob sich der Status MEDIUM ONLINE (MOL) setzen und rücksetzen läßt, während die Wartungsgerätebereitschaft vorliegt. Dazu setzt der Test den Diagnosemodus und liest das Gerätestatusregister RMDS. Es wird erwartet, daß das Bit MOL im Register RMDS Null ist.Ist das nicht der Fall, erfolgt Fehler 1.
Anschließend wird dann die Wartungsgerätebereitschaft gesetzt und erwartet, daß das Bit MOL im Register RMDS gesetzt ist.Wurde es nicht gesetzt, kommt es zum Fehler 2.
Traten bei dem Test bis hierher keine Fehler auf,wird geprüft,ob VOLUME VALID (Datenträger gültig) gelöscht ist. Ist es nicht rückgesetzt, erfolgt Fehler 3.
Ansonsten wird mit verschiedenen Datenmustern die Bit-Interferenz getestet. Dabei kann Fehler 4 auftreten.

6.6.6.25. Test 25: WRITE LOCK TEST Schreibschutztest

Es wird geprüft, ob sich der Status WRITE LOCK (WRL) setzen und rücksetzen läßt. Dazu wird der Diagnosemodus eingeschaltet und geprüft, ob WRL gelöscht ist. Ist das nicht der Fall, erfolgt Fehler 1.
Anschließend wird Wartungsschreibschutz gesetzt und geprüft, ob das Bit WRL gesetzt ist. Wurde es jedoch nicht gesetzt, kommt es zum Fehler 2.

6.6.6.26. Test 26: DRIVE FAULT TEST Gerätefehlertest

Der Test prüft, ob sich DEVICE CHECK (DVC) und UNSAFE (UNS) setzen und rücksetzen lassen, während der Wartungsgerätefehler (MDF) gesetzt ist.

Mit Diagnosemodus setzen, setzt das Programm das Bit MDF im Register RMMR1. Die Register RMER1 und RMER2 werden dann gelöscht und es wird geprüft, ob DVC gelöscht ist. Wurde DVC nicht rückgesetzt, erfolgt Fehler 1. Das Register RMER2 wird ausgelesen um zu prüfen, ob das Bit DVC gesetzt ist. Das Register RMER1 wird ebenfalls gelesen, dabei muß das Bit UNS gesetzt sein. Bei Nichtübereinstimmung, kommt es entweder zum Fehler 2 oder Fehler 3.
Danach wird MDF gelöscht und geprüft, ob dadurch DVC und UNS rückgesetzt wurden.

6.6.6.27. Test 27: SEEK ERROR TEST Positionierfehler (SKI) - Test

Der Test prüft, ob sich der SEEK-ERROR setzen bzw. rücksetzen läßt.
Als erstes wird dazu der Diagnosemodus gesetzt und geprüft, ob das Bit SKI im Fehlerregister 2 gelöscht ist. Ansonsten kommt es zum Fehler 1.
Anschließend wird das Bit MSER im Wartungsregister 1 gesetzt und geprüft,ob sich dadurch das Bit SKI setzen ließ.
Traten bis hierher keine Fehler auf, erfolgt noch eine Kontrolle auf Bit-Interferenz durch Testmuster. In Abhängigkeit, ob im Testmuster das entsprechende Bit für MSER gesetzt ist, wird das Bit SKI ausgewertet. Ist es nicht in Ordnung, kommt es zum Fehler 2.

6.6.6.28. Test 28: PIP TEST Positionierung in Ausführung (PIP) - Test

Es wird geprüft, ob sich der Status PIP setzen bzw. rücksetzen läßt, während Diagnosemodus eingeschaltet ist.
Der Diagnosemodus und das Bit MOC im Register RMMR1 werden gesetzt,danach muß das Bit PIP im Register RMDS Null sein. Ansonsten erfolgt Fehler 1.
Das Bit MOC wird anschließend rückgesetzt und PIP muß dann auf Eins stehen. Wurde es nicht gesetzt, kommt es zum Fehler 2.
Traten bis hierher keine Fehler auf, erfolgt eine Kontrolle auf Bit-Interferenz durch Testmuster. In Abhängigkeit, ob im Testmuster das entsprechende Bit für MOC gesetzt ist, wird das Bit PIP ausgewertet. Ist es nicht in Ordnung, kommt es zum Fehler 3.

6.6.6.29. Test 29: EBL TEST Test auf Status Ende des Blocks (EBL)

Der Test prüft, ob sich der Status Ende des Blockes (EBL) setzen bzw. rücksetzen läßt.
Als erstes wird kontrolliert, ob der Status EBL gelöscht ist, sonst kommt es zum Fehler 1.
Anschließend schaltet das Programm den Diagnosemodus ein und prüft, ob das Bit EBL noch rückgesezt ist. Sonst erfolgt Fehler 2.
Danach wird DEBL gesetzt und geprüft, ob das Bit EBL gesetzt ist. Wurde es nicht gesetzt, kommt es zum Fehler 3.
Traten bis hierher keine Fehler auf, erfolgt eine Kontrolle auf Bit-Interferenz durch Testmuster. In Abhängigkeit, ob im Testmuster das entsprechende Bit für DEBL gesetzt ist, wird das Bit EBL ausgewertet. Ist es nicht in Ordnung, kommt es zum Fehler 4.

6.6.6.30. Test 30: LAST SECTOR, LAST TRACK TEST Letzter Sektor (LS),letzte Spur (LST) - Test

In diesem Test wird die Sektor/Spur Platenadresse geprüft.
Der Test schreibt dazu alle möglichen Muster in das Plattenadreßregister RMDA und prüft den Status LAST SECTOR und LAST SECTOR/TRACK für jedes Muster. Sind die Bits LS und LST nicht korrekt für die entsprechende Plattenadresse, kommt es entweder zum Fehler 1 oder Fehler 2.
Der Test wird ausgefürht sowohl für 16-bit-Format,wie auch für 18-bit-Format.

6.6.6.31. Test 31: RMDA COUNT TEST RMDA Zähltest

Der Test prüft, ob sich das Plattenadreßregister RMDA richtig inkrementieren läßt.
Dazu erhöht der Test das Register RMDA unter Benutzung von DIAGNOSE END OF BLOCK (DEBL) und prüft das Resultat im 16- und im 18-bit-Format.
Als erstes wird dazu kontinuierlich der Sektorzähler von 0 bis zur Spuradressenerhöhung inkrementiert. Wird der Sektorzähler dabei nicht richtig erhöht, kommt es zum Fehler 1.
Als zweites wird der Spurzähler inkrementiert. Er beginnt bei Spur 0, letzter Sektor. Es wird ein kompletter Spurzyklus gezählt. Wird der Spurzähler nicht richtig erhöht, kommt es zum Fehler 2.

6.6.6.32. Test 32: RMDC COUNT TEST RMDC Zähltest

Der Test prüft, ob sich das Zylinderadreßregister RMDC richtig inkrementieren läßt.
Dazu erhöht das Programm das Register RMDC unter Benutzung von DIAGNOSE END OF BLOCK und prüft, ob das Register RMDC in einem kompletten Zyklus erhöht wurde. Konnte die Zylinderadresse nicht richtig erhöht werden,kommt es zum Fehler 1.

6.6.6.33. Test 33: LBT TEST Letzten Block nehmen (LBT)

Der Test prüft, ob der Status LAST BLOCK TAKEN (LBT) gelöscht ist, wenn das Register RMDA beschrieben wird. Ist das nicht der Fall, kommt es zum Fehler 1. Außerdem wird geprüft, ob LBT gesetzt ist, wenn der letzte Sektor transportiert ist. Ansonsten kommt es zum Fehler 2.

6.6.6.34. Test 34: COMPOSITE ERROR TEST Sammelfehler (ERR) - Test

Der Test benutzt die Initialisierung und den Diagnosemodus um alle Fehler rückzusetzen. Dann wird geprüft, ob ERR=0. Ansonsten kommt es zum Fehler 1.
Anschließend werden beide Fehlerregister mit Einsen geladen und geprüft, ob ERR gesetzt ist. Ist der Sammelfehler danach nicht gesetzt, kommt es zum Fehler 2.
Danach wird geprüft, ob ERR gesetzt ist,für jedes gesetzte Bit im Fehlerregister 1. Sonst kommt es zum Fehler 3.

Zum Schluß wird geprüft, ob ERR gesetzt ist,für jedes gesetzte Bit im Fehlerregister 2. Ist ERR jedoch nicht gesetzt, kommt es zum Fehler 4.

6.6.6.35. Test 35: WRITE GO TEST Schreiben GO - Test

Der Test prüft, ob sich das GO-Bit setzen läßt.
Dazu wird eine NOP-Funktion und das GO-Bit in das Register RMCS1 transportiert. Es werden alle Funktionskods getestet. Es erfolgt Fehler 1, wenn sich GO nicht mit dem Funktionscode setzen läßt.

6.6.6.36. Test 36: BRANCH MULTIPLEXOR TEST Verzweigungsmultiplexor - Test

Dieser Test prüft den Ausgang des Verzweigungsmultiplexers.
Als erstes wird dazu die Testtaktfreigabe gesetzt. Das Bit TST im Register RMMR2 muß danach Eins sein (Fehler 1).
Anschließend werden verschiedene Funktionen geladen und das TST-Bit ausgewertet. Wenn das TST-Bit für die entsprechende Funktion nicht korrekt ist,kommt es zum Fehler 2.

6.6.6.37. Test 37: SET/RESET GO TEST Setzen/Rücksetzen GO

Dieser Test prüft, ob sich das GO-Bit setzen und rücksetzen läßt.
Dazu ist das Subsystem initialisiert und der Diagnosemodus eingestellt mit gesetztem Bit DEBUG CLOCK ENABLE im Register RMMR1.Bestimmte Funktionscodes werden in das Register RMCS1 geschrieben und das Programm liest dann das Register RMCS1 aus, um zu prüfen, ob das GO-Bit gesetzt ist. Wurde es dabei nicht gesetzt, kommt es zum Fehler 1.Das Register RMDS wird ebenfalls ausgelesen um zu prüfen, ob das Bit DRY (Gerät bereit) rückgesetzt ist. Ansonsten kommt es zum Fehler 2.
Anschließend benutzt das Programm den Testtakt und prüft,ob GO gelöscht und DRY gesetzt ist. Ist das nicht der Fall, kommt es zum Fehler 3,Fehler 4 oder Fehler 5.

6.6.6.38. Test 38: END 1 RESET GO TEST END 1 Rücksetzen GO - Test

Der Test prüft, ob der COMMAND SEQUENCER das GO-Bit rücksetzen kann.
Der Test arbeitet dazu die Funktionen RELEASE,SEARCH und eine illegale Funktion (Funktionscode 32) im Diagnosemodus ab. Zum Fehler 1 kommt es, wenn nach Laden des Funktionscodes einschließlich GO-Bit, das GO-Bit nicht gesetzt ist. Fehler 2 erfolgt, wenn das GO-Bit rückgesetzt ist, bevor der Taktzähler nicht Null ist. Zum Fehler 3 kommt es, wenn GO nach bestimmten Taktzyklus nicht rückgesetzt wurde.

6.6.6.39. Test 39: SET PULSE TEST
Setzimpuls - Test

Der Test prüft,ob der COMMAND SEQENCER den Setzimpuls generieren kann. Als erstes wird dazu die Testtaktfreigabe gesetzt. Danach muß der Status CONTINUE im Register RMMR1 gelöscht sein (Fehler 1).
Anschließend werden verschiedene Funktionen geladen und der Status CONTINUE ausgewertet. Ist der Status für die entsprechende Funktion nicht korrekt, kommt es zum Fehler 2.

6.6.6.40. Test 40: SET/RESET IVC TEST
Setzen/Rücksetzen IVC
(ungültiges Kommando)

Bei diesem Test wird für jeden Funktionscode der Status IVC ausgetestet.
Als erstes wird kontrolliert,ob nach INIT der Status IVC gelöscht ist. Ansonsten kommt es zum Fehler 1.
Anschließend wird der Funktionscode und GO in das Register RMCS1 geladen. Jeder Funktionscode wird getestet und das Bit IVC im Register RMER2 wird kontrolliert. Ist der Status IVC nicht korrekt, erfolgt Fehler 2.

6.6.6.41. Test 41: SET LOSS OF SYSTEM CLOCK
(LSC) - TEST
Setzen Status LSC (Ausfall
Systemtakt)

Der Test prüft, ob sich das Bit LSC setzen bzw. rücksetzen läßt.
Als erstes wird kontrolliert, ob nach INIT der Status LSC gelöscht ist. Ansonsten kommt es zum Fehler 1.
Anschließend wird der Testtakt freigegeben und das GO-Bit gesetzt. Danach wird genügend Zeit gewartet zum Setzen des Monoflops.
Der Test inaktiviert dann den Testtakt und prüft, ob das Bit LSC gesetzt ist. Ist das nicht der Fall, kommt es zum Fehler 2.

6.6.6.42. Test 42: DECODE TEST
Decodiertest

Der Test prüft, ob das DECODE Flop auf dem IF-Modul mit der Vorderflanke des SET PULSE gesetzt wird ,außer wenn COMPOSITE ERROR aktiv ist.
Der Test benutzt dazu VOLUME VALID und OCCUPIED um zu bestimmen, ob das DECODE Flop gesetzt oder rückgesetzt ist. Durch Initialisierung sind VV und OCC rückgesetzt.
Das Programm führt dann solche Kommandos aus, welche VV und OCC setzen. Anschließend wird geprüft, ob eins oder beide Bits gesetzt sind.
Die selben Kommandos werden dann mit gesetztem COMPOSITIVE ERROR (Sammelfehler) ausgeführt und es wird kontrolliert, ob keins von beiden Bits gesetzt ist.

6.6.6.43. Test 43: SET/RESET VOLUME VALID
(VV) TEST
Setzen/Rücksetzen VV
(Datenträger gültig)

Der Test prüft, ob der Status VOLUME VALID mit der Vorderflanke von UNIT READY rückgesetzt wird, und gesetzt mit den Sofort-Kommandos PACK ACKNOWLEDGE und READ

IN PRESET.
Der Test führt dazu einen Null/Eins-Übergang der UNIT READY (Gerätebereitschaft) aus und prüft, ob der Status VV Null ist. Ist er nicht Null, kommt es zum Fehler 1.
Anschließend führt das Programm die Kommandos PACK ACKNOWLEDGE und READ IN PRESET aus und kontrolliert, ob dadurch der Status VV gesetzt wurde.
Konnte es durch diese Funktionen nicht gesetzt werden, erfolgt Fehler 2.

6.6.6.44. Test 44: ILLEGAL FUNCTION (ILF) TEST
Test auf illegale Funktion

Bei diesem Test wird auf Status ILF (illegale Funktion) geprüft.
Als erstes wird geprüft, ob VOLUME VALID (Datenträger gültig) gesetzt ist. Ansonsten kommt es zum Fehler 1.
Das Programm prüft dann, ob das Bit ILF des Registers RMER1 bei legalen Funktionen nicht gesetzt und bei illegalen Funktionen gesetzt ist. Das GO-Bit wird dabei ignoriert. Ist der Status dabei nicht korrekt, kommt es zum Fehler 2.

6.6.6.45. Test 45: OCCUPIED (OCC) TEST
Belegungstest

Der Test prüft, ob das Bit OCC während Datentransporte gesetzt ist und bei allen anderen Kommandos nicht gesetzt ist.
Als erstes wird geprüft,ob VOLUME VALID (Datenträger gültig) gesetzt ist. Ansonsten kommt es zum Fehler 1.
Danach wird der Testtakt freigegeben um vorzubeugen, daß GO rückgesetzt wird, bevor der Status abgefragt ist. Ist der Status OCC dabei nicht korrekt, kommt es zum Fehler 2.

6.6.6.46. Test 46: READ IN PRESET (RIP) TEST
Lesevorbereitungstest

Der Test setzt jeden Status oder die Registerbits, die durch das READ IN PRESET-Kommando gelöscht werden.
Anschließend wird das Kommando READ IN PRESET ausgeführt und geprüft, ob ein oder mehrere Bits gelöscht sind. Im einzelnen werden folgende Tests ausgeführt.

- Die Bits 10 bis 12 im Register RMOF werden durch eine MOVE-Instruktion gesetzt. Der Test ist beendet, wenn diese Bits nach Ausführung des Kommandos RIP wieder gelöscht sind.
- Das Zylinderadreßregister wird durch eine MOVE-Instruktion gesetzt. Der Test ist beendet, wenn die Bits 00...03 oder 04...07 oder 08...09 auf Null rückgesetzt sind, nach Ausführung des Kommandos RIP.
- Das Plattenadreßregister wird durch eine MOVE-Instruktion gesetzt. Der Test ist beendet, wenn die Bits 00...03 oder 04...07 oder 08...11 oder 12...15 auf Null rückgesetzt sind, nach Ausführung des Kommandos RIP.

Wurde keins der 3 Register gelöscht, kommt es zum Fehler 1.

6.6.6.47. Test 47: READ IN PRESET (RIP)/ RMOF TEST
Lesevorbereitung/
Offsetregister - Test

Der Test prüft, ob durch das Kommando READ IN PRESET die Bits HCI, ECI und FMT16 im Register RMOF rückgesetzt werden. Die Bits HCI,ECI und FMT werden dazu im

Offsetregister gesetzt.
Danach wird das Kommando READ IN PRESET ausgeführt und jedes dieser Bits muß danach auf Null rückgesetzt sein. Wurden sie jedoch nicht gelöscht, kommt es zum Fehler 1.

6.6.6.48. Test 48: RMDA/RMDC/READ IN PRESET (RIP) TEST RMDA/RMDC/Lesevorbereitung - Test

Der Test prüft, ob das Kommando RIP die Zylinderadresse und die Plattenadresse rücksetzt. Die Register RMDC und RMDA werden dazu geladen.
Anschließend wird das Kommando RIP ausgeführt, um beide Register zu löschen. Wurde das Register RMDA nicht rückgesetzt, kommt es zum Fehler 1. Fehler 2 wird angezeigt, wenn das Register RMDC nicht gelöscht werden konnte.

6.6.6.49. Test 49: OFFSET COMMAND TEST Test des Offsetkommandos

Der Test prüft, ob sich mit Hilfe des Kommandos OFFSET der OFFSET-MODUS (OM) setzen läßt.
Als erstes wird geprüft, ob der Status VOLUME VALID (Datenträger gültig) gesetzt werden konnte. Ist das nicht der Fall, erfolgt Fehler 1.
Anschließend führt der Test das Kommando OFFSET aus und prüft, ob danach das Bit OM im Register RMDS gesetzt ist. Ansonsten kommt es zum Fehler 2.

6.6.6.50. Test 50: RETURN TO CENTER TEST Bewegen auf Spurmitte

Der Test prüft, ob das Kommando RETURN TO CENTER den OFFSET-MODUS (OM) ausschaltet.
Dazu wird der Status OM im Register RMDS, durch das Kommando OFFSET eingeschaltet. Konnte er nicht gesetzt werden, kommt es zum Fehler 2.
Dann wird das Kommando RETURN TO CENTER ausgeführt und geprüft, ob der Status OM dadurch ausgeschaltet wurde. Ist das nicht der Fall, erfolgt Fehler 4.
Fehler 1 oder Fehler 3 werden gemeldet, wenn vor Ausführung der beiden Kommandos, der Status VOLUME VALID (Datenträger gültig) nicht gesetzt wurde.

6.6.6.51. Test 51: RMDC CLEAR OFFSET TEST Löschen OFFSET MODUS (OM) durch RMDC

Der Test prüft, ob der Status OM gelöscht ist, nachdem eine Zylinderadresse eingetragen wurde.
Als erstes wird dabei der Status VV (Datenträger gültig) geprüft. Ist er nicht gesetzt, kommt es zum Fehler 1.
Anschließend wird das Kommando OFFSET ausgeführt. Wurde dabei der Status OM nicht gesetzt, kommt es zum Fehler 2.
Danach wird das Zylinderadreßregister beschrieben und geprüft, ob dadurch das Bit OM im Register RMDS gelöscht wird. Wurde es nicht rückgesetzt, erfolgt Fehler 3.

6.6.6.52. Test 52: EBL CLEAR OFFSET TEST
END OF BLOCK (EBL) löscht
OFFSET MODUS (OM)

Der Test prüft, ob sich durch eine Kopfumschaltung der Status OM löschen läßt.
Als erstes wird geprüft,ob sich der Status VV (Datenträger gültig) setzen läßt.
Ist das nicht der Fall, erfolgt Fehler 1.
Anschließend wird das Kommando OFFSET ausgeführt, um den OFFSET-MODUS einzustellen. Wurde er nicht gesetzt, kommt es zum Fehler 2.
Danach wird das Format-Bit gesetzt und die letzte Kopf-Sektoradresse in das Register RMDA geladen. Der Test bewirkt EBL und prüft, ob der Status OM gelöscht ist. Ansonsten kommt es zum Fehler 3.

6.6.6.53. Test 53: RUN AND GO TEST
Test auf Status RUN AND GO

Es wird geprüft, ob der Status RUN AND GO während Lese-und Schreibkommandos gesetzt ist.
Dazu wurde ein Datentransportkommando mit GO-Bit ins Register RMCS1 geschrieben.
Das Bit RUN AND GO im Register RMMR1 muß dann für jedes Datenkommando auf Eins gesetzt sein. Der Testtakt ist freigegeben, so daß GO nicht rückgesetzt ist, bevor der Status abgefragt wurde. Zum Fehler 1 kommt es, wenn der Status RUN AND GO für den entsprechenden Funktionskode nicht korrekt ist.

6.6.6.54. Test 54: SET IAE TEST
Test auf falsche
Adresse (IAE)

Der Test prüft, ob bei falscher Adressenangabe das Fehlerbit IAE gesetzt wird.
Als erstes wird geprüft, ob sich der Status VV (Datenträger gültig) setzen läßt.
Ist das nicht der Fall, erfolgt Fehler 1.
Anschließend lädt das Programm eine ungültige Sektor-, Kopf- und Zylinderadresse und führt unterschiedliche Kommandos aus. Es wird dann geprüft, ob das Bit IAE im Register RMER1 gesetzt ist. Wurde es dabei nicht gesetzt, kommt es zum Fehler 2.

6.6.6.55. Test 55: SEARCH,SEEK,READ,WRITE TEST
Suchen, Positionieren,
Lesen, Schreiben

Der Test prüft, ob die SEARCH, SEEK, READ und WRITE-Entschlüsselung des IF-Moduls korrekt ist für alle Kommandos.
Zuerst wird der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend wird eine ungültige Sektor,-Kopf-und Zylinderadresse geladen. Das Programm führt dann jedes Kommando aus und testet dabei auf IAE-Bit (falsche Adresse). Ist es dabei nicht korrekt, kommt es zum Fehler 2.

6.6.6.56. Test 56: INVALID SECTOR/TRACK TEST
Test auf ungültige Sektor/
Spuradresse

Der Test prüft, ob eine ungültige Kopf-Sektoradresse festgestellt werden kann.
Als erstes wird dazu der Status VV (Datenträger gültig) gesetzt. Konnte er nicht

gesetzt werden, kommt es zum Fehler 1.
Anschließend wird ein Testmuster in das Register RMDA geladen und das Kommando SEARCH ausgeführt. Es wird dann geprüft, ob das Bit IAE (falsche Adresse) im Fehlerregister 1 gesetzt wurde. Ist der Status IAE nicht gesetzt, kommt es zum Fehler 2.
Der Test wird mit verschiedenen Testmustern wiederholt.

6.6.6.57. Test 57: INVALID CYLINDER TEST Test auf ungültige Zylinderadresse

Der Test prüft, ob eine ungültige Zylinderadresse festgestellt werden kann.
Zuerst wird dazu der Status VV (Datenträger gültig) gesetzt. Ließ er sich nicht setzen, erfolgt Fehler 1.
Das Programm lädt dann ein Testmuster in das Register RMDC und führt das Kommando SEARCH aus. Es wird anschließend geprüft, ob das Bit IAE (falsche Adresse) im Fehlerregister 1 gesetzt ist. Konnte es nicht gesetzt werden, kommt es zum Fehler 2.
Der Test wird mit verschiedenen Testmustern wiederholt.

6.6.6.58. Test 58: SET AOE TEST Test auf Adressenüberlauf (AOE)

Der Test prüft, ob Adressenüberlauf festgestellt werden kann.
Als erstes wird dabei der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Das Programm lädt danach die letztmögliche Kopf-Sektoradresse in das Register RMDA, und in das Register RMDC die letztmögliche Zylinderadresse. Anschließend wird ein Datenkommando mit Testtaktfreigebe ausgelöst. Das Programm prüft dann, ob dabei das Bit AOE im Register RMER1 gesetzt wurde. Wurde es nicht gesetzt, kommt es zum Fehler 2.

6.6.6.59. Test 59: SET RMR TEST Test RMR (Registermodifikation verweigern)

Der Test prüft, ob das Bit RMR im Fehlerregister 1 gesetzt wird, wenn ein Register bei gesetztem GO-Bit beschrieben wird, außer wenn das Achtung- oder Wartungsregister beschrieben werden. Hierzu wird der Testtakt freigegeben. Dann wird das NOP-Kommando und das GO-Bit in das Register RMCS1 geschrieben.
Anschließend beschreibt das Programm das Wartungsregister und das Register RMAS. Hierbei darf das Fehlerbit RMR nicht gesetzt sein (Fehler 1).
Danach wird das Register RMDA beschrieben und es wird kontrolliert, ob der Status RMR gesetzt wurde (Fehler 2).

6.6.6.60. Test 60: PGM STATUS CHECK Kontrolle auf Status PGM (programmierbar)

Der Test dient zur Prüfung, ob das PROGRAMMABLE (PGM) Status-Bit und das DRIVE REQUEST (Geräteanforderung) Status-Bit kompatibel sind.
Das Programm teilt Fehler 1 mit, wenn der Status PGM gesetzt ist und DRQ nicht gesetzt ist.

PGM ist nicht vorhersagbar in dem Fall, wenn der Status DRQ gesetzt ist.

6.6.6.61. Test 61: DPR STATUS CHECK Kontrolle auf Status DPR (Gerät vorhanden)

Dieser Test prüft, ob der Status DPR (Gerät vorhanden) und der Status DVA (Gerät verfügbar) gesetzt sind.
Zum Fehler 1 kommt es, wenn DVA nicht gesetzt ist. Fehler 2 wird mitgeteilt, wenn das Bit DPR nicht gesetzt ist.

6.6.6.62. Test 62: PORT REQUEST TEST - PART 1 Kanalanforderungstest - Teil 1

Der Test prüft, ob die Kanalanforderungs-Bits gesetzt sind, wenn das Programm das Register RMCS1 ausliest.
Als erstes wird dabei der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Danach führt der Test das Kommando RELEASE (Freigabe) aus. In der Annahme das ein Kanal freigegeben ist, wird dann das Register RMCS1 gelesen.
Anschließend wird das Register RMMR2 gelesen und geprüft, ob eins der Kanalanforderungs-Bits gesetzt ist. Wurde jedoch keins gesetzt, kommt es zum Fehler 2.

6.6.6.63. Test 63: PORT REQUEST TEST - PART 2 Kanalanforderungstest - Teil 2

Der Test prüft, ob die Kanalanforderungs-Bits gesetzt sind,wenn das Programm das Register RMAS beschreibt.
Als erstes wird dabei der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Danach führt der Test das Kommando RELEASE (Freigabe) aus und das Register RMAS wird beschrieben.
Anschließend wird das Register RMMR2 ausgelesen und geprüft, ob eins der Kanalanforderungs-Bits gesetzt ist. Wurde jedoch keins gesetzt, kommt es zum Fehler 2.

6.6.6.64. Test 64: PORT REQUEST TEST - PART 3 Kanalanforderungstest - Teil 3

Der Test prüft, ob die Kanalanforderung gesetzt ist,wenn irgendein Register außer RMAS beschrieben wird.
Als erstes wird dabei der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Danach beschreibt das Programm das Register RMDA und prüft, ob ein Kanalanforderungs-Bit gesetzt ist. Ist das nicht der Fall, kommt es zum Fehler 2.

5.6.6.65. Test 65: RELEASE TEST
Freigabe - Test

Dieser Test prüft, ob das Kommando RELEASE die Anforderungs-Bits rücksetzt. Als erstes wird dazu der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend führt der Test das Kommando RELEASE im Wartungsmodus aus. Es wird danach geprüft, ob die Bits RQA und RGB im Register RMMR2 gelöscht sind. Ist das nicht der Fall, kommt es zum Fehler 2.

6.6.6.66. Test 66: WRITE ATA TEST
Schreiben/Achtung
(ATA) - Test

Dieser Test prüft, ob das Bit ATA im Register RMDS gelöscht wird, wenn das Register RMAS beschrieben wird. Das Programm setzt dazu MAINTENANCE UNIT READY, welches ein Achtung hervorruft.
Anschließend wird das Register RMAS beschrieben. Es wird dann geprüft, ob das Bit ATA im Register RMDS gelöscht ist. Ansonsten kommt es zum Fehler 1. Fehler 2 wird mitgeteilt, wenn die Bits ATA im Register RMAS nicht gelöscht sind.

6.6.6.67. Test 67: RESET ATA BY GO TEST
Rücksetzen Achtung
(ATA) durch GO

Der Test prüft, ob das Bit ATA gelöscht wird, bei gesetztem GO-Bit und gelöschtem Sammelfehlerbit. Das Programm setzt MAINTENANCE UNIT READY, welches ein Achtung hervorruft.
Anschließend wird mit Testtaktfreigebe das GO-Bit gesetzt. Das Bit ATA im Register RMDS muß danach rückgesetzt sein. Konnte es nicht gelöscht werden, kommt es zum Fehler 1.

6.6.6.68. Test 68: UNIT READY ATA TEST
Achtung durch UNIT READY
(Gerät bereit)

Dieser Test prüft, ob Null/Eins- und Eins/Null Übergänge von UNIT READY, das Bit ATA im Register RMDS setzen. Der Test wird im Diagnosemodus ausgeführt, um beide Übergänge von UNIT READY zu erzeugen, und zu prüfen, ob ATA mit jedem Übergang gesetzt wird. Zum Error 1 kommt es, wenn ATA beim Null/Eins-Übergang nicht gesetzt ist. Fehler 2 wird angezeigt beim Eins/Null-Übergang.

6.6.6.69. Test 69: ERROR ATA TEST
Achtung (ATA) setzen durch
Sammelfehler

Dieser Test prüft, ob ATA gesetzt ist, wenn Sammelfehler auftritt, während das GO-Bit nicht mehr aktiv ist.
Das Programm löscht das Gerät und lädt das Register RMER1 mit Einsen.
Anschließend prüft es, ob das Bit ATA im Register RMDS gesetzt ist. Ist das nicht der Fall, kommt es zum Fehler 1.

6.6.6.70. Test 70: REGISTER TRANSFER ATA TEST
Achtung durch Registerübertragung

Der Test prüft, ob das Bit ATA gesetzt ist, wenn irgendein Register außer RMAS und RMCS beschrieben wird, während Sammelfehler gesetzt ist.
Dazu lädt das Programm das Register RMER1 mit Einsen. Das Achtung wird danach gelöscht.
Anschließend werden die Register RMAS und RMCS beschrieben , und es wird kontrolliert, ob das Bit ATA nicht gesetzt ist. Ansonsten erfolgt Fehler 1.
Danach wird das Register RMDC beschrieben und geprüft, ob das Bit ATA gesetzt ist. Wurde es nicht gesetzt, kommt es ebenfalls zum Fehler 1.

6.6.6.71. Test 71: POSITIONING SET ATA TEST
Achtung (ATA) setzen durch Positionierung

Dieser Test prüft, ob die Kommandos OFFSET und RETURN TO CENTERLINE das Bit ATA im Register RMDS setzen.
Als erstes wird dazu der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend werden die Kommandos OFFSET und RETURN TO CENTERLINE ausgeführt. Nach jedem Kommando wird kontrolliert, ob das Bit ATA im Register RMDS gesetzt wurde. Ansonsten kommt es zum Fehler 2.

6.6.6.72. Test 72: SET WLE TEST
Test auf Setzen Schreibschutzfehler (WLE)

Der Test setzt zuerst den Status VV (Datenträger gültig). Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend prüft der Test im einzelnen:
- ob bei eingeschaltetem WRITE PROTECT (Schreibschutz) und Anweisung des Kommandos WRITE DATA das Bit WLE im Register RMER1 gesetzt wird
- ob bei gelöschtem WRITE PROTECT und Anweisung des Kommandos WRITE DATA das Bit WLE gelöscht ist
- ob bei gesetztem WRITE PROTECT und Anweisung des Kommandos READ DATA das Bit WLE gelöscht ist
- ob bei eingeschaltetem WRITE PROTECT und Anweisung des Kommandos READ IN PRESET das Bit WLE gelöscht ist

Zum Fehler 2 kommt es, wenn der Status WLE nicht korrekt ist.

6.6.6.73. Test 73: EXCEPTION TEST
Ausnahmetest

Dieser Test prüft, ob nach Initialisierung das Bit EXCEPTION im Register RMMR1 rückgesetzt ist. Wurde es nicht gelöscht, kommt es zum Fehler 1.
Als nächstes wird der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, wird Fehler 2 mitgeteilt.
Anschließend wird das Kommando WRITE DATA angewiesen. Die Register RMDA und RMDC wurden dazu vorher gelöscht.
Zum Schluß wird das Bit RMR im Register RMER1 gesetzt, und es erfolgt die Kontrolle, ob das Bit EXCEPTION gesetzt wurde. Ist das nicht der Fall, kommt es zum Fehler 3.

6.6.6.74. Test 74: RECALIBRATE TEST Nullstellen

Das Kommando RECALIBRATE wird durch 6 Untertests geprüft:
- Prüfung ob OPI gesetzt wird, wenn die Gerätebereitschaft ausfällt. Wurde OPI nicht gesetzt, kommt es zum Fehler 2.
- Prüfung ob das Kommando RECALIBRATE abbricht, wenn eine Abbruchbedingung vorliegt. Ansonsten kommt es zum Fehler 4.
- Prüfung ob sich OPI setzen läßt, wenn sich der Zylinder Latch nicht löschen läßt. Wurde OPI nicht gesetzt, kommt es zum Fehler 6.
- Prüfung ob ATA gesetzt ist, wenn ein komplettes Gerätenullstellen ausgeführt wurde. Ansonsten kommt es zum Fehler 8.
- Prüfung ob das Kommando RECALIBRATE nach Ausführung, während einer Warteschleife abbricht. Zum Fehler 10 kommt es dabei, wenn GO zu früh rückgesetzt wurde. Fehler 11 wird mitgeteilt, wenn GO nicht rückgesetzt wurde.
- Prüfung des TAG-Bus, während Kommando RECALIBRATE. Ist der Zustand dabei nicht korrekt, kommt es zum Fehler 13.

Bei jedem Teiltest wird außerdem zuerst der Status VV gesetzt. Konnte er nicht gesetzt werden, kommt es entsprechend dem Teiltest, entweder zum Fehler 1, Fehler 3, Fehler 5, Fehler 7, Fehler 9 oder Fehler 12.

6.6.6.75. Test 75: SEEK TEST Positionierung

Das Kommando SEEK wird mit folgenden Untertest getestet:
- Prüfung ob OPI gesetzt wird, wenn Gerätebereitschaft ausfällt. Wurde OPI nicht gesetzt, kommt es zum Fehler 2.
- Prüfung ob das Kommando SEEK abbricht, wenn eine Abbruchbedingung vorliegt. Ansonsten kommt es zum Fehler 4.
- Prüfung ob OPI gesetzt ist, wenn sich der Zylinder Latch nicht löschen läßt. Wurde OPI nicht gesetzt, kommt es zum Fehler 6.
- Prüfung ob ATA gesetzt ist, wenn das Gerät eine komplette Positionierung ausgeführt hat. Ansonsten kommt es zum Fehler 8.
- Prüfung ob das Kommando SEEK nach Ausführung, während einer Warteschleife abbricht. Zum Fehler 10 kommt es dabei, wenn GO zu früh rückgesetzt wurde. Fehler 11 wird mitgeteilt, wenn GO nicht rückgesetzt wurde.
- Prüfung des TAG Bus während des Kommandos SEEK. Ist der Zustand dabei nicht korrekt, kommt es zum Fehler 13.

Bei jedem Teiltest wird zuerst der Status VV gesetzt. Konnte er nicht gesetzt werden, kommt es entweder zum Fehler 1, Fehler 3, Fehler 5, Fehler 7, Fehler 9 oder Fehler 12.

6.6.6.76. Test 76: SEARCH TEST Suchkommando - Test

Das Kommando SEARCH wird mit folgenden Untertests getestet:
- Prüfung ob OPI gesetzt wird, wenn die Gerätebereitschaft ausfällt. Wurde OPI nicht gesetzt, kommt es zum Fehler 2.
- Prüfung ob das Kommando SEARCH abbricht, wenn eine Abbruchbedingung vorliegt. Ansonsten kommt es zum Fehler 4.
- Prüfung ob OPI gesetzt ist, wenn sich der Zylinder Latch nicht löschen läßt. Wurde OPI nicht gesetzt, kommt es zum Fehler 6.
- Prüfung ob ATA gesetzt ist, wenn das Gerät ein komplettes Kommando SEARCH ausgeführt hat. Ansonsten kommt es zum Fehler 8.
- Prüfung ob das Kommando SEARCH abbricht, während einer Suchpositionierschleife. Zum Fehler 10 kommt es dabei, wenn GO zu früh

rückgesetzt wurde. Fehler 11 wird mitgeteilt, wenn GO nicht rückgesetzt wurde.
- Prüfung ob das Kommando SEARCH abbricht, während einer Sektorvergleichsschleife. Zum Fehler 13 kommt es, wenn SEARCH ENABLE nicht gesetzt ist. Fehler 14 wird mitgeteilt, wenn GO nicht rückgesetzt wurde.
- Prüfung des TAG Bus während des Kommandos SEARCH. Ist der Zustand dabei nicht korrekt, kommt es zum Fehler 16.

Bei jedem Teiltest wird zuerst der Status VV gesetzt. Konnte er nicht gesetzt werden, kommt es entweder zum Fehler 1, Fehler 3, Fehler 5, Fehler 7, Fehler 9, Fehler 12 oder Fehler 15.

6.6.6.77. Test 77: SEARCH TIMEOUT TEST Suchkommando Time-Out - Test

Der Test prüft, ob OPI gesetzt ist, bei SEARCH TIMEOUT.
Als erstes wird dazu der Status VV (Datenträger gültig) gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend erfolgt Testtaktfreigabe und das Kommando SEARCH wird geladen und gestartet. Wenn sich der Seqenzer in der Sektorvergleichsschleife befindet, wird kontolliert, ob der Status SEARCH ENABLE gesetzt ist. Ist das nicht der Fall, kommt es zum Fehler 2.
Fehler 3 wird gemeldet, wenn der Status OPI bei SEARCH TIMEOUT nicht gesetzt ist.

6.6.6.78. Test 78: DATA COMMAND TESTS (1) Datenkommandotest 1

Der Datenkommandotest 1 ist in 5 Untertests gegliedert:
- Prüfung ob OPI gesetzt wird, wenn das Gerät nicht bereit ist. Wurde es nicht gesetzt, kommt es zum Fehler 2.
- Prüfung ob das Datenkommando abbricht, wenn eine Abbruchbedingung vorliegt. Wenn der Status EBL nicht gesetzt ist, kommt es zum Fehler 4. Läßt sich der Status EBL nicht wieder rücksetzen, erfolgt Fehler 5. Fehler 6 wird mitgeteilt, wenn sich GO nicht rücksetzen läßt innerhalb 4 Taktzyklen.
- Prüfung ob der Sequenzer zur Positionierung springt, wenn die RUN-Leitung aktiv ist. Zum Fehler 8 kommt es, wenn der Status RUN AND GO nicht gesetzt ist. Fehler 9 wird mitgeteilt, wenn der TAG Bus während des Datenkommandos nicht korrekt ist.
- Prüfung ob das Datenkommando abbricht bei Sequenzerposition 144,145. Wurde RUN AND GO nicht gesetzt, kommt es zum Fehler 11. Fehler 12 wird mitgeteilt, wenn das Datenkommando nicht abbricht, bei Gerätefehler. (EBL wurde nicht gesetzt)
- Prüfung auf HEAD TAG während Datenkommando. Zum Fehler 14 kommt es, wenn RUN AND GO nicht gesetzt sind. Fehler 15 wird mitgeteilt, wenn der TAG Bus während des Datenkommandos nicht korrekt ist.

Bei jedem Teiltest wird zuerst der Status VV gesetzt. Konnte er nicht gesetzt werden, kommt es entweder zum Fehler 1, Fehler 3, Fehler 7, Fehler 10 oder Fehler 13.

6.6.6.79. Test 79: DATA COMMAND TESTS (2) Datenkommandotest 2

Der Datenkommandotest 2 ist in 5 Untertests gegliedert:

- Prüfung ob OPI gesetzt wird, wenn ON CYLINDER LATCH sich nicht rücksetzen läßt. Zum Fehler 2 kommt es, wenn RUN AND GO nicht gesetzt werden konnte. Fehler 3 wird mitgeteilt, wenn OPI nicht gesetzt wurde.
- Prüfung ob das Datenkommando abbricht, während der Positionierschleife. Zum Fehler 5 kommt es, wenn RUN AND GO nicht gesetzt werden konnte. Zum Fehler 6 kommt es, wenn das Datenkommando nicht abbricht, (EBL nicht gesetzt) bei gesetztem MSER.
- Prüfung ob das Datenkommando abbricht, während OFFSET, wenn ON CYLINDER LATCH. Zum Fehler 8 kommt es dabei, wenn der OFFSET-MODUS durch das Kommando OFFSET nicht eingeschaltet werden konnte. Fehler 9 wird mitgeteilt, wenn RUN AND GO nicht gesetzt wurde. Zum Fehler 10 kommt es, wenn OPI nicht gesetzt wurde.
- Prüfung ob das Datenkommando abbricht, während Offsetwarteschleife. Fehler 12 wird mitgeteilt, wenn OFFSET-MODUS nicht gesetzt wurde. Zum Fehler 13 kommt es, wenn RUN AND GO nicht gesetzt wird. Fehler 14 erscheint, wenn das Datenkommando nicht abbricht, bei gesetztem Gerätefehler.
- Prüfung ob das Datenkommando während Sektorwarteschleife abbricht. Zum Fehler 16 kommt es, wenn RUN AND GO nicht gesetzt ist. Wurde ESRC (Suchfreigabe) im Register RMMR1 nicht gesetzt, kommt es zum Fehler 17. Fehler 18 wird mitgeteilt,wenn das Datenkommando bei vorliegender Gerätefehlerbedingung nicht abbricht.

6.6.6.80. Test 80: DATA COMMAND TESTS (3)
Datenkommandotest 3

Bei diesem Test wird der TAG Bus während des Datenkommandos getestet. Der Test ist in zwei Testabschnitte unterteilt.
Im ersten Teil wird dazu eine Vorwärtspositionierung benutzt. Im zweiten Teil eine Rückwärtspositionierung. Beide Testabschnitte benutzen die gleiche Subroutine.
Als erstes wird dabei der Status VV gesetzt. Konnte er nicht gesetzt werden, kommt es zum Fehler 1.
Anschließend werden Spur-, Sektor-, und Zylinderadresse geladen und der entsprechende Offset eingetragen. Wurde dabei der OFFSET-MODUS nicht gesetzt, kommt es zum Fehler 2.
Danach wird der Wortzähler geladen, der Testtakt freigegeben und das Datenkommando geladen.
Zum Fehler 3 kommt es, wenn RUN AND GO nicht gesetzt sind. Fehler 4,5 und 6 werden gemeldet, wenn der Tag Bus nicht korrekt ist während des Datenkommandos.

6.6.7. Auflistung der Tests

Test 1: PROBE. DEVICE REGISTERS
Geräteregister ansprechen
Test 2: QUALIFICATION TESTS
Qualifikationsprüfung
Test 3: CTOD TEST
Kontroller zum Gerät - Test
Test 4: MSBUS INITIALIZE TEST
MSBUS Initialisierungstest
Test 5: CLEAR STUCK ACTIVE TEST
Löschen ständig aktiv
Test 6: TRISTATE TRANSFER TEST
Tristate - Übertragungstest
Test 7: REGISTER SELECT TEST
Registerauswahltest

Test 8: DRIVE TYPE TEST
Gerätetyp - Test
Test 9: DEVICE AVAILABLE TEST
Gerät verfügbar - Test
Test 10: HOLDING REGISTER TRANSFER TEST
Halteregister (RMHR) - Übertragungstest
Test 11: CONTROL STATUS #1 TRANSFER TEST
Steuerstatusregister 1 (RMCS1) - Test
Test 12: ERROR REGISTER 1 TRANSFER TEST
Fehlerregister 1 (RMER1) - Übertragungstest
Test 13: ERROR REGISTER 2 TRANSFER TEST
Fehlerregister 2 (RMER2) - Übertragungstest
Test 14: CLEAR OFFSET STUCK ACTIVE TEST
Offsetlöschen ständig aktiv
Test 15: OFFSET REGISTER TRANSFER TEST
Offsetregister (RMOF) - Übertragungstest
Test 16: SERIAL NUMBER TEST
Seriennummernregister (RMSN) - Test
Test 17: CONTROL BUS PARITY GENERATION
Prüfzeichenlogiktest
Test 18: RMDA,RMDC FAULT TEST
RMDA,RMDC Fehlertest
Test 19: DISK ADDRESS TRANSFER TEST
Plattenadressen - Übertragungstest
Test 20: DESIRED CYLINDER TRANSFER TEST
Zylinderadressen - Übertragungstest
Test 21: ILLEGAL REGISTER TEST
Test auf illegales Register
Test 22: RESET GO BY INIT TEST
Rücksetzen GO durch INIT
Test 23: DIAGNOSTIC MODE TEST
Diagnosemodus - Test
Test 24: MEDIUM ONLINE TEST
Gerät online - Test
Test 25: WRITE LOCK TEST
Schreibschutztest
Test 26: DRIVE FAULT TEST
Gerätefehlertest
Test 27: SEEK ERROR TEST
Positionierfehler (SKI) - Test
Test 28: PIP TEST
Positionierung in Ausführung (PIP) - Test
Test 29: EBL TEST
Test auf Status Ende des Blocks (EBL)
Test 30: LAST SECTOR, LAST TRACK TEST
Letzter Sektor (LS),letzte Spur (LST) - Test
Test 31: RMDA COUNT TEST
RMDA - Zähltest
Test 32: RMDC COUNT TEST
RMDC - Zähltest
Test 33: LBT TEST
Letzten Block nehmen (LBT
Test 34: COMPOSITE ERROR TEST
Sammelfehler (ERR) - Test
Test 35: WRITE GO TEST
Schreiben GO - Test
Test 36: BRANCH MULTIPLEXOR TEST
Verzweigungsmultiplexor - Test

Test 37: SET/RESET GO TEST
Setzen/Rücksetzen GO
Test 38: END 1 RESET GO TEST
END 1 Rücksetzen GO - Test
Test 39: SET PULSE TEST
Setzimpuls - Test
Test 40: SET/RESET IVC TEST
Setzen/Rücksetzen IVC (ungültiges Kommando)
Test 41: SET LOSS OF SYSTEM CLOCK (LSC) - Test
Setzen Status LSC (Ausfall Systemtakt)
Test 42: DECODE TEST
Decodiertest
Test 43: SET/RESET VOLUME VALID (VV) TEST
Setzen/Rücksetzen VV (Datenträger gültig)
Test 44: ILLEGAL FUNCTION (ILF) TEST
Test auf illegale Funktion
Test 45: OCCUPIED (OCC) TEST
Belegungstest
Test 46: READ IN PRESET (RIP) TEST
Lesevorbereitungstest
Test 47: READ IN PRESET (RIP)/RMOF TEST
Lesevorbereitung/Offsetregister - Test
Test 48: RMDA/RMDC/READ IN PRESET (RIP) TEST
RMDA/RMDC/Lesevorbereitung - Test
Test 49: OFFSET COMMAND TEST
Test des Offsetkommandos
Test 50: RETURN TO CENTER TEST
Bewegen auf Spurmitte
Test 51: RMDC CLEAR OFFSET TEST
Löschen OFFSET MODUS (OM) durch RMDC
Test 52: EBL CLEAR OFFSET TEST
END OF BLOCK (EBL) löscht OFFSET MODUS (OM)
Test 53: RUN AND GO TEST
Test auf Status RUN AND GO
Test 54: SET IAE TEST
Test auf falsche Adresse (IAE)
Test 55: SEARCH,SEEK,READ, WRITE TEST
Suchen,Positionieren,Lesen,Schreiben
Test 56: INVALID SECTOR/TRACK TEST
Test auf ungültige Sektor/Spuradresse
Test 57: INVALID CYLINDER TEST
Test auf ungültige Zylinderadresse
Test 58: SET AOE TEST
Test auf Adressenüberlauf (AOE)
Test 59: SET RMR TEST
Test RMR (Registermodifikation verweigern)
Test 60: PGM STATUS CHECK
Kontrolle auf Status PGM (programmierbar)
Test 61: DPR STATUS CHECK
Kontrolle auf Status DPR (Gerät vorhanden)
Test 62: PORT REQUEST TEST - PART 1
Kanalanforderungstest - Teil 1
Test 63: PORT REQUEST TEST - PART 2
Kanalanforderungstest - Teil 2
Test 64: PORT REQUEST TEST - PART 3
Kanalanforderungstest - Teil 3
Test 65: RELEASE TEST
Freigabe - Test

Test 66: WRITE ATA TEST
Schreiben/Achtung (ATA) - Test
Test 67: RESET ATA BY GO TEST
Rücksetzen Achtung (ATA) durch GO
Test 68: UNIT READY ATA TEST
Achtung durch UNIT READY (Gerät bereit)
Test 69: ERROR ATA TEST
Achtung (ATA) setzen durch Sammelfehler
Test 70: REGISTER TRANSFER ATA TEST
Achtung durch Registerübertragung
Test 71: POSITIONING SET ATA TEST
Achtung (ATA) setzen durch Positionierung
Test 72: SET WLE TEST
Test auf Setzen Schreibschutzfehler (WLE)
Test 73: EXCEPTION TEST
Ausnahmetest
Test 74: RECALIBRATE TEST
Nullstellen
Test 75: SEEK TEST
Positionierung
Test 76: SEARCH TEST
Suchkommando - Test
Test 77: SEARCH TIMEOUT TEST
Suchkommando Time-Out - Test
Test 78: DATA COMMAND TESTS (1)
Datenkommandotest 1
Test 79: DATA COMMAND TESTS (2)
Datenkommandotest 2
Test 80: DATA COMMAND TESTS (3)
Datenkommandotest 3

6.7. PVRGA: Formatiertest

6.7.1. Zusammenfassung

Das Diagnoseprogramm PVRGA (Dok.-Nr. 1.57.700607.2) ist ein Programm der Ebene 2. Das Programm formatiert den PM80. Es wird unter Supervisor entweder Stand-alone oder unter SVP 1800 aufgerufen.

6.7.2. Testbedingungen

6.7.2.1. Hardware

Die minimale Hardwarekonfiguration besteht aus:

- RVS K 1840 (mit Minimalkonfiguration)
- Speicher MSA40
- Konsolterminal
- PM80

Das Gerät muß sich in Stellung online befinden.

6.7.2.2. Software

PVRGA benötigt den Diagnose-Supervisor PSSAA stand-alone oder in Verbindung mit SVP 1800. Auch für die Arbeit ohne Betriebssystem wird der Driver PVQDR.EXE benötigt.

6.7.3. Voraussetzungen

Für eine Abarbeitung des Programms wird die volle Funktionstüchtigkeit aller Elemente der CPU vorausgesetzt.

6.7.4. Bedienungshinweise

Die Bedienungsanleitung des Diagnose-Supervisors ist für den Lade-und Startvorgang mitzuverwenden.
Der Diagnose-Supervisor meldet sich mit:

```
DS>
```

Das Programm PVRGA ist zu laden:

```
DS> LOAD PVRGA.EXE
```

Vor dem Programmstart muß das Testgerät angeschlossen (ATTACH) und für den Test logisch ausgewählt werden (SELECT).

```
DS> ATTACH RHA40
DEVICE LINK? HUB
DEVICE NAME? RHn
TR? number? (dezimal 1-15) Standard <8>
BR? level? (dezimal 4-7) Standard <5>
DS> SELECT RHn
DS> ATTACH PM80
DEVICE LINK? RHn
DEVICE NAME? DRAn
DS> SELECT DRAn
```

Die Übertragungsstufe (TR) und die Busanforderungsstufe (BR) sind zusätzliche Informationen für eine korrekte Geräteeingliederung.

Programmstart:

```
DS> START/Schalter
```

Weitere Bedienungshinweise sind der Diagnose-Supervisor-Beschreibung zu entnehmen, z.B. die Auswahl bestimmter Testsektionen (s. Abschn. 6.7.13.) über den Diagnose-Supervisor.

6.7.5. Funktionsbeschreibung

6.7.5.1. Übersicht

Das Programm PVRGA beinhaltet 7 Tests. Eine Auflistung der Tests ist in Abschnitt 6.7.8. zu finden.
Das Programm wird in zwei Objekt-Module unterteilt:

Modul 1: enthält Datenbasis und die Subroutinen
Modul 2: enthält Test 1-7

6.7.5.2. Programmgröße

Der Speicherbedarf für das Programm beträgt 41472. byte (81 Pages).

6.7.5.3. Programmlaufzeit

	Stand-alone	User-Modus
Normal	20 min	30 min
Schnellauf	08 min	11 min

6.7.5.4. Abarbeitungsbedingungen

entfällt

6.7.5.5. Flag-Verwendung

Ereignisflag 23

Dieses Flag bewirkt, daß die Anfangsplattenadresse für jeden Schritt in der Sektion ausgegeben wird.
In der Sektion FORMAT ist das dann die Adresse der Spur, welche formatiert wird.
In der Sektion VERIFY, ist das die Adresse der getesteten Spuren.

Ereignisflag 22

Dieses Flag bewirkt die Ausgabe der Anfangsplattenadresse jeweils einmal aller 100 Zylinder. Dieses Flag hebt das Flag 23 auf.

Ereignisflag 21

Das Flag 21 bewirkt, daß Fehlermitteilungen im Kurzformat ausgegeben werden.

Ereignisflag 20

Das Flag ist nur effektiv im Stand-alone. Wenn dieses Flag gesetzt ist, wird jeweils eine Spur bei der Plattenprüfung verarbeitet.

Ereignisflag 19

Wenn dieses Flag gesetzt ist, werden nur die FE-Zylinder formatiert. Das Schreiben auf den Nutzerbereich wird verhindert.

Ereignisflag 18

Wenn das Flag 18 gesetzt ist und die Sektion FORMAT gestartet wurde, wird die Platte ohne Prüfung formatiert.

Ereignisflag 2

Das Flag 2 erlaubt Systemfehlerprotokollierung. Dieses Flag ist nur im Nutzermodus effektiv.

Ereignisflag 1

Das Flag 1 erlaubt QIO-Fehlerwiederholung. Bei normaler Nutzung sollte dieses Flag nicht gesetzt sein.

QUICK

Das Quick-Flag wird nur während der Sektion VERIFY benutzt. Unter normalen Bedingungen wird jede Spur 5 * gelesen. Wenn das Flag QUICK gesetzt wird, wird jede Spur nur einmal gelesen.

6.7.5.6. Allgemeine Programmanforderungen

entfällt

6.7.5.7. Fehlermitteilung

Das Fehlerausschriftformat besteht aus 2 oder 3 Teilen. Der erste Teil ist der zweizeilige Fehlerkopf. Der zweite Teil besteht aus dem Fehlerstatuscode des Registers R0 (HEX), dem Prozeß, der ihn meldet (SVP, PSSAA oder PVRGA) und einer kurzen Definition. Im wahlweise dritten Teil werden die Geräteregister ausgegeben. Die Ausgabe der Geräteregister wird von dem IE3-Flag gesteuert. Die Ausgabe des Statuscodes wird von dem IE2-Flag gesteuert. Die Ausgabe der Fehlermitteilung wird von dem IE1-Flag gesteuert.

Gerätefehlerformat

Ein Gerätefehler ist jeder Nichtdatenfehler, der im Gerät oder dem Kanal, an dem das Gerät angegliedert ist, auftritt. Das Format besteht aus einem zweizeiligen Fehlerkopf, dem Fehlerstatuscode, den Adapterregistern, den Geräteregistern, dem QIO-Wiederhol- und -Maximalzähler und den QIO-Start/Stopp-Zeiten.
Ist Ereignisflag 21 gesetzt, werden nicht die Adapterregister, sondern nur der Plattenstatus, Fehler- und Plattenadreßregister ausgeschrieben.
Der QIO-Wiederholzähler ist 0, wenn Ereignisflag 1 gelöscht ist. Der QIO-Maximalzähler enthält die Maximalzahl der QIO-Wiederholungen bei Fehler, wenn Ereignisflag 1 gesetzt ist.
QIO-Start/Stopp-Zeiten sind die Start- und Stoppzeiten des QIO, in dem ein Fehler auftritt.

Beispiel-Gerätefehlerausschrift (Kurzformat Ereignisflag 21=1):

```
******** PVRGA PM80/IDC P80 DISK FORMATTER - 1.0 ********
Pass 1, test 3, subtest 0, error 8, 15-JUN-1987 04:47:38.00
System fatal error while testing DRA0: RETRY FAILURE

STATUS RETURN      PROCESS      DEFINITION
------------------------------------------------------------
00AC00F0(X)        PVRGA        RETRY COUNT EXHAUSTED

REGISTER    CONTENTS        BIT DEFINITIONS
------------------------------------------------------------
RMCS1   :       0806(X) ; DVA,FUNCTION=RECALIBRATE
RMDS    :       13C0(X) ; MOL,PGM,DPR,DRY,VV
RMER1   :       0000(X) ;
RMER2   :       0000(X) ;
RMDC    :       0000(X) ; DESIRED CYLINDER=00000(D)
RMDA    :       0001(X) ; TRACK=00(D),SECTOR=01(D)
RMOF    :       1000(X) ; FMT16
RMLA    :       07C0(X) ; SECTOR=31(D)

FINAL ERROR COUNT=   0
MAX   ERROR COUNT=   8

QIO STOP  TIME= 04:47:37.99
QIO START TIME= 04:47:37.60

********  End of System fatal error number 8  ********

 .. Aborted program at pass 1, test 3, subtest 0, PC 0000958A
```

Beispiel-Gerätefehlerausschrift (Langformat Ereignisflag 21=0):

```
******** PVRGA PM80/IDC P80 DISK FORMATTER - 1.0 ********
Pass 1, test 3, subtest 0, error 8, 15-JUN-1987 04:27:36.64
System fatal while testing DRA0: RETRY FAILURE

STATUS RETURN       PROCESS         DEFINITION
---------------------------------------------------------
00AC00F0(X)         PVRGA           RETRY COUNT EXHAUSTED

REGISTER     CONTENTS           BIT DEFINITIONS
---------------------------------------------------------
MBACSR   :    00000020(X) ; ADAPTER CODE=20(X)
MBACR    :    00000004(X) ; IE
MBASR    :    00000000(X) ;
MBAVAR   :    00000020(X) ; MAP POINTER=00(X),
                          ; PAGE BYTE ADDRESS=0020(X)
MBABCR        BF80BF98(X) ; MSBUS BYTE COUNT=BF80(X),
                          ; ISB BYTE COUNT=BF98(X)
MBAFMAP  :    80000356(X) ; VALID,PFN=000356(X)
MBAPMAP  :    00000000(X) ; PFN=000000(X)
RMCS1    :        0806(X) ; DVA,FUNCTION=RECALIBRATE
RMDS     :        13C0(X) ; MOL,PGM,DPR,DRY,VV
RMER1    :        0000(X) ;
RMER2    :        0000(X) ;
RMDC     :        0000(X) ; DESIRED CYLINDER=00000(D)
RMDA     :        0001(X) ; TRACK=00(D),SECTOR=01(D)
RMOF     :        1000(X) ; FMT16
RMLA     :        07C0(X) ; SECTOR=31(D)
RMMR1    :        0068(X) ; ECCE,MWR,MSCLK
RMMR2    :        93FF(X) ; REQA,TEST,BUS IN LINES=3FF(X)
RMDT     :        2816(X) ; MOH,DRQ,DRIVE TYPE=PM80
RMAS     :        0000(X) ;
RMSN     :        2286(X) ; SERIAL NUMBER=2286(X)
RMEC1    :        0836(X) ; BURST LOCATION=0836(X)
RMEC2    :        0000(X) ; ERROR BURST=0000(X)
RMRH     :        0000(X) ;

FINAL ERROR COUNT=    0
MAX   ERROR COUNT=    8

QIO STOP  TIME= 04:27:36.64
QIO START TIME= 04:27:36.24

********  End of System fatal error number 8  ********

 .. Aborted program at pass 1, test 3, subtest 0,PC 0000958A
```

Systemservicefehlerformat

Ein Systemservicefehler ist jeder Fehler, der entweder durch den Diagnose-Supervisor oder durch das Betriebssystem festgestellt wurde, während der FORMATTER eine Serviceroutine anfordert (z.B. ein $DS_GETBUF).

Beispiel einer Systemservicefehlermitteilung:

```
******** PVRGA PM80 FORMATTER - 1.0 ********
PASS 0 INITIALIZATION SECTION ERROR 5 15-JUN-1987 11:54:38.02
SYSTEM FATAL WHILE TESTING DRA0:   FAILED TO OBTAIN BUFFER MEMORY

STATUS RETURN        PROCESS        DEFINITION
-------------------------------------------------------------
00000244             SVP            VIRTUAL ADDRESS SPACE FULL

 .. ABORTED PROGRAM AT PASS 0 INITIALIZATION SECTION PC 00005ED7
```

Datenfehlermeldung der READALL-Sektion

```
********** READALL DATA ERROR REPORT **********

TRACK=0(D)

                              TIMES          ERROR   LISTED IN BAD/SKIP
  CYLINDER      SECTOR        RETRIED        CODE    SECTOR FILE
---------------------------------------------------------------------------
      95(D)        11(D)         10(D)        UNECC       NO
     112(D)         4(D)          5(D)        ECC         NO
     157(D)        18(D)         10(D)        BSE         YES

TRACK=4(D)

                              TIMES          ERROR   LISTED IN BAD
  CYLINDER      SECTOR        RETRIED        CODE    SECTOR FILE
---------------------------------------------------------------------------
      18(D)         3(D)         10(D)        UNECC       NO
     295(D)        21(D)         10(D)        HCRC        NO
```

Fehlercodeerläuterung

CODE	DEFINITION
ECC	korrigierbarer Datenfehler
UNECC	unkorrigierbarer Datenfehler
BSE	Bit 15 oder 14 des Kopfwortes sind gelöscht
HCRC	Kopf CRC-Fehler
HCE	Kopfvergleichsfehler
FER	Formatfehler
OPI	Operation unvollständig

6.7.5.8. Programmitteilungen

Die Übersichtsausschrift enthält den Plattentyp, den Plattenerkennungsnamen und entsprechend der aufgerufenen Sektion, die Anzahl der Sektoren, die zur BAD- oder SKIP-Sektordatei addiert wurden.
Die Sektion READALL gibt die Anzahl der defekten Sektoren aus.

6.7.5.9. Leistungsparameter des Testobjektes

entfällt

6.7.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.7.5.11. Testfolge bei mehreren Einheiten

entfällt

6.7.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.7.5.13. Testsektionen

Das Programm PVRGA enthält 9 Testsektionen, wobei jede durch einen Sektionsschalter in der Startkommandozeile ausgewählt werden kann.

DEFAULT

Test 7

Diese Sektion gibt nur eine Liste der Sektionsnamen und deren Funktion aus.

DISPLAY

Test 2

Diese Sektion wird benutzt, um den Inhalt der SKIP- und BAD-Sektordatei anzuzeigen. Außerdem prüft diese Sektion die Gültigkeit der Dateien. Diese Sektion gibt sowohl die physische Adresse des defekten Sektors, als auch die logische Blocknummer aus.

FORMAT

Test 2-4

Diese Sektion liest und bestätigt beide Sektordateien. Wenn sich die Datei nicht lesen läßt, wird das Programm abgebrochen. Nach erfolgreichem Lesen, wird die Platte spurweise formatiert.
Zur Komplettierung der FORMAT-Sektion wird eine Plattenprüfung ausgeführt. Jeder dabei festgestellte defekte Sektor wird gekennzeichnet und in die BAD- oder SKIP-Sektordatei eingetragen.

Dies ist abhänging vom Fehlertyp und der Position. Der Home-Block wird im PSEUDO ODS2-Format, mit dem Kennzeichen SKRATCH beschrieben.

HELP

Test 7

Diese Sektion gibt nur eine Liste der Sektionsnamen und deren Funktion aus.

HDAINI

Test1-4

Diese Sektion fragt als erstes den Nutzer, ob die BAD- oder SKIP-Sektordatei zu initialisieren ist. Sobald dies festgestellt wurde, wird die Spur, auf der sich die entsprechende Datei befindet, formatiert und mit Nulleintragungen versehen.

Achtung: Diese Sektion sollte nur aufgerufen werden, wenn das Plattensubsystem funktionstüchtig ist, und die BAD- oder SKIP-Sektordatei zerstört sind.

READALL

Test 2,5

Diese Sektion liest jeden Sektor von der Platte und gibt jeden gefundenen Datenfehler in einen HEAD-MAP ERROR REPORT aus. Auf einem als in Ordnung bekannten Gerät kann die Sektion benutzt werden zum Suchen von Datenfehlern auf der Platte.

REBUILD

Test 1,2,6

Diese Sektion ermöglicht es dem Nutzer, die BAD- oder SKIP-Sektordatei wiederherzustellen, falls eine zerstört wurde. Als erstes wird die Spur, auf der die ausgewählte Datei lokalisiert ist, formatiert. Anschließend wird die Datei mit Nulleintragungen versehen. Ab hier verläuft dann der Test analog der Sektion UPDATE.

UPDATE

Test 2,6

Diese Sektion ermöglicht dem Nutzer die manuelle Aktualisierung der BAD- oder SKIP-Sektordatei. Die Sektion fordert vom Nutzer die Adresse des defekten Blocks ab. Nachdem der Nutzer alle defekten Blöcke ausgewählt hat, werden sie gekennzeichnet. Danach schreibt das Programm die BAD- oder SKIP-Sektordatei zurück.

Achtung: Diese Sektion sollte nicht benutzt werden, es sei denn, der Nutzer ist vertraut mit der SKIP-Sektorierung des PM80 und weiß Bescheid über die Struktur der BAD-Sektordatei. Die Daten auf der Platte werden nicht gerettet.

VERIFY

Test 2,4

Diese Sektion liest und bestätigt beide Sektordateien. Wenn sich die Datei nicht lesen läßt, wird das Programm abgebrochen. Nach erfolgreichem Lesen wird eine Plattenprüfung durchgeführt. Jeder dabei festgestellte defekte Sektor wird gekennzeichnet und in die BAD- oder SKIP-Sektordatei eingetragen. Dies ist abhängig vom Fehlertyp und der Position des Fehlers. Der Home-Block wird im PSEUDO ODS2-Format, mit dem Kennzeichen SKRATCH beschrieben.

6.7.6. Testbeschreibungen

6.7.6.1. Test 1: INITIALIZE BAD AND/OR SKIP SECTOR FILE Initialisierung BAD-und/oder SKIP-Sektordatei

Dieser Test initialisiert die BAD- und/oder SKIP-Sektordatei. Diese Dateien werden mit Nullen beschrieben. Bevor initialisiert wird, werden die Spuren auf denen die Dateien lokalisiert sind, formatiert.

Testschritte:

1. Beginn
2. Wenn PROTECTION nicht Null, dann EXIT
3. Registerladen
4. Abfrage der Plattennummer
5. Abfrage, ob BAD-Sektordatei initialisiert werden soll. Bei Nein, Sprung zur Abfrage SKIP-Sektordatei.
6. Warnmeldung
7. Abfrage, ob weitere Abarbeitung. Bei Nein, Abbruch
8. Initialisierung BAD-Sektordatei
9. Wenn Fehler, dann Fehlerausschrift und Testabbruch
10. Auswertung IE23-Flag. Wenn gesetzt, Ausgabe der Plattenadresse
11. Löschen Fehler-Flag für BAD-Sektordatei
12. Abfrage, ob SKIP-Sektordatei initialisiert werden soll. Bei Nein, EXIT
13. Warnmeldung
14. Abfrage nach weiterer Abarbeitung. Bei Nein, Abbruch
15. Initialisierung SKIP-Sektordatei
16. Wenn Fehler, dann Fehlerausschrift und Testabbruch
17. Auswertung IE23-Flag. Wenn gesetzt, Ausgabe der Plattenadresse
18. Löschen Fehler-Flag für SKIP-Sektordatei
19. Ende

6.7.6.2. Test 2: DISPLAY BAD AND SKIP SECTOR FILE Anzeigen BAD- und SKIP-Sektordatei

Dieser Test liest und prüft die BAD- und SKIP-Sektordatei auf ihre Gültigkeit. Die Dateien stehen im Speicher im logischen Format.

Testschritte:

1. Beginn
2. Wenn PROTECTION = 1, dann EXIT
3. Registerladen

4. Lesen BAD-Sektordatei
5. Wenn Fehler, dann Auswertung, ob weicher oder harter Fehler vorliegt.
   Es folgt entsprechende Fehlerausschrift.
6. Anzeige BAD-Sektordatei
7. Lesen SKIP-Sektordatei
8. Wenn Fehler, dann Auswertung, ob weicher oder harter Fehler vorliegt.
   Es folgt entsprechende Fehlerausschrift.
9. Anzeige SKIP-Sektordatei
10. Ende

6.7.6.3. Test 3: FORMAT HDA
Formatieren Platte

Dieser Test formatiert die Platte und kennzeichnet sie im PSEUDO ODS2 - Format als freie Arbeitsplatte (SKRATCH).
Dieser Test erfordert eine gültige BAD/SKIP-Sektordatei.

Testschritte:

1. Beginn
2. Wenn Schreibschutz, dann Testabbruch
3. Wenn PROTECTION = 1, dann EXIT
4. Registerladen
5. Ausgabe Startzeit
6. Aufbereiten Spur
7. Wenn Fehler, dann Fehlerlöschen und Wiederholungen. Weiterhin Fehler, dann Abbruch.
8. Bilden Adreßworte für eine Spur
9. Formatieren Spur
10. Wenn Fehler, dann Fehlerlöschen und Wiederholungen. Weiterhin Fehler, dann Abbruch.
11. Sind alle Spuren formatiert? Wenn nicht, Wiederholung ab Schritt 6
12. Kennzeichnen Platte
13. Ausgabe Stoppzeit
14. Ende

6.7.6.4. Test 4: SURFACE VERIFICATION
Plattenprüfung

Dieser Test prüft die Platte. Jeder Defekt, der festgestellt wird, wird gekennzeichnet. Am Ende dieses Test wird die BAD- und SKIP-Sektordatei aktualisiert und zurückgeschrieben. Die Platte wird gekennzeichnet im PSEUDO ODS2 Format als freie Arbeitsplatte (SKRATCH).
Der Test verlangt eine gültige BAD- und SKIP-Sektordatei.

Testschritte:

1. Beginn
2. Wenn PROTECTION = 1, dann EXIT
3. Wenn FATAL = 1, Warnung: BAD- und SKIP-Sektordatei sind nicht lesbar
4. Prüfflag:=1
5. Registerladen
6. Ausgabe Startzeit
7. Wenn QUICK = 1, dann nur einen Grenzfall benutzen
8. Testen Spur
9. Wenn Fehler, Kontrolle auf Datenfehler. Bei Datenfehler, Suchen des

defekten Sektors. Bei Nichtdatenfehler, Fehlerlöschen und Wiederholung.
10. Alle Grenzfälle abgearbeitet? Wenn nicht, weiter bei Schritt 8.
11. Alle Spuren getestet? Wenn nicht, weiter bei Schritt 8.
12. Liegen Eintragungen für BAD-Sektordatei vor? Bei Nein, zu Schritt 15.
13. Kennzeichnen Spur
14. Noch weitere Eintragungen vorhanden? Bei Ja, weiter bei Schritt 13.
15. Liegen Eintragungen für SKIP-Sektordatei vor? Bei Nein,zu Schritt 18.
16. Kennzeichnen Spur
17. Noch weitere Eintragungen vorhanden? Bei Ja, weiter ab Schritt 16.
18. Kennzeichnen Platte
19. Ausgabe der BAD- und/oder SKIP-Sektordatei entsprechend ob Eintragungen erfolgten.
20. Ausgabe Stoppzeit
21. Ende

6.7.6.5. Test 5: READ HDA
Lesen Platte

Dieser Test liest jeden Sektor auf der Platte. Es wird jeder defekte Sektor, der nicht gekennzeichnet ist, in die BAD- oder SKIP-Sektordatei eingetragen.

Testschritte:

1. Beginn
2. Wenn PROTECTION = 1, dann EXIT
3. Registerladen
4. Ausgabe Startzeit
5. Lesen Spur
6. Wenn Fehler, Kontrolle auf Datenfehler. Bei Ja, retten Fehlerstelle. Bei Nein, Testwiederholung.
7. Wurden alle Spuren gelesen? Bei Nein Wiederholung ab Schritt 5.
8. Ende

6.7.6.6. Test 6: REBUILD BAD OR SKIP SECTOR FILE
Wiederherstellen BAD- oder
SKIP-Sektordatei

Dieser Test fordert vom Bediener die Angabe welche Datei zu erneuern ist, wenn sie zerstört ist.

Testschritte:

1. Beginn
2. Aufforderung zur Eingabe, welche Datei zu erneuern ist
3. Wenn keine, dann Testabbruch
4. Registerladen
5. Aufforderung zur Eingabe des Eingabeformats (logisch oder physisch)
6. Wenn logisch, Eingabe der logischen Blocknummer
7. Eintragung in BAD- oder SKIP-Sektordatei
8. Wenn noch mehr Eintragungen gewünscht, weiter bei Schritt 6. Ansonsten weiter bei Schritt 12.
9. Wenn physisch, Eingabe von Zylinder, Spur und Sektor
10. Eintragung in BAD- oder SKIP-Sektordatei
11. Wenn noch mehr Eintragungen gewünscht, weiter bei Schritt 9. Ansonsten weiter bei Schritt 12.

6.8. PVRGB: Festplattenspeicher Funktionstest

6.8.1. Zusammenfassung

Das Programm PVRGB (Dok.-Nr. 1.57.700608.0) ist ein Programm der Ebene 3. Es besteht aus einer Testserie, welche die ordnungsgemäße Beschaffenheit des PM80 nachweist. Das Programm setzt voraus, daß der Test PVRDA fehlerfrei gelaufen ist.

6.8.2. Testbedingungen

6.8.2.1. Hardware

Die minimale Hardwarekonfiguration besteht aus:

- RVS K 1840 (mit Minimalkonfiguration)
- RHA40 und einem oder mehreren PM80

Das Gerät muß sich in Stellung online befinden. Die Systemfehlerprotokollierung wird nicht unterstützt.

6.8.2.2. Software

- Diagnose-Supervisor
- Das Programm PVRGB und der Diagnose-Supervisor müssen vor dem Start geladen sein.

6.8.3. Voraussetzungen

Für eine Abarbeitung des Programms wird die volle Funktionsfähigkeit aller Elemente der CPU vorausgesetzt. Der Diagnose-Supervisor muß geladen sein.

6.8.4. Bedienungshinweise

Die Bedienungsanleitung des Diagnose-Supervisors ist für den Lade-und Startvorgang mitzuverwenden.
Der Diagnose-Supervisor meldet sich mit:

```
DS>
```

Das Testprogramm PVRGB ist zu laden:

```
DS> LOAD PVRGB.EXE
```

Vor dem Programmstart muß das Testgerät angeschlossen (ATTACH) und für den Test logisch ausgewählt werden (SELECT).

```
DS> ATTACH RHA40
DEVICE LINK? HUB
DEVICE NAME? RHn
TR? number? (dezimal 1-15) Standard <8>
BR? level?  (dezimal 4-7)  Standard <5>
DS> SELECT RHn
DS> ATTACH PM80
DEVICE LINK? RHn
DEVICE NAME? DRAn
DS> SELECT DRAn
```

Die Übertragungsstufe (TR) und die Busanforderungsstufe sind zusätzliche Informationen für eine korrekte Geräteeingliederung.

Programmstart:

```
DS> START/Schalter
```

Weitere Bedienungshinweise sind der DS-Beschreibung zu entnehmen.

6.8.5. Funktionsbeschreibung

6.8.5.1. Übersicht

Das Programm beinhaltet 40 Tests. Eine Auflistung der Tests ist im Abschnitt 6.8.8. zu finden.
Das Programm wird in fünf Objekt-Module unterteilt:

Modul 1:	enthält globale Daten und Subroutinen
Modul 2:	Test 1-13
Modul 3:	Test 14-20
Modul 4:	Test 21-34
Modul 5:	Test 35-40

6.8.5.2. Programmgröße

Der Speicherbedarf für dieses Programm beträgt 55296. bytes, 108. Pages.

6.8.5.3. Programmlaufzeit

Die Programmlaufzeit für einen fehlerfreien Standarddurchlauf beträgt ca. 1:38 min.

6.8.5.4. Abarbeitungsbedingungen

entfällt

6.8.5.5. Flagverwendung

entfällt

6.8.5.6. Allgemeine Programmanforderungen

entfällt

6.8.5.7. Fehlerermittlung

Das Programm PVRGB liefert umfangreiche Fehlermeldungen. Die erste Zeile der Fehlermeldung enthält die Durchlaufzahl, die Testnummer, die Subtestnummer und die Fehlernummer. Dieser Zeile folgt die eigentliche Fehlermitteilung.
An dem nebenstehenden Beispiel soll eine Fehlermitteilung erläutert werden.

```
(1)  ******** PVRGB - PM80 FUNCTIONAL DIAGNOSTIC -1.0 ********
(2)  Pass 1, test 14, subtest 2, error 6, 21-JUL-1987 11:26:49.86
(3)  Hard error while testing DRA0: SECTOR DID NOT INCIMENT ON A SKIP SECTOR ERROR

(4)  REGISTER               DATA          MNEMONICS

     RMCS1 [60010400]:      0838(X)       ;DVA,FUNCTION=READ DATA
     RMDS  [60010404]:      1BC0(X)       ;MOL,WRL,PGM,DPR,DRY,VV
     RMER1 [60010408]:      0000(X)       ;
     RMMR  [6001040C]:      0028(X)       ;MWR,MSCLK
     RMAS  [60010410]:      0000(X)       ;
     RMDA  [60010414]:      0001(X)       ;TRACK=00(D),SECTOR=01(D)
     RMDT  [60010418]:      2816(X)       ;MOH,DRQ,DRIVE TYPE = PM80
     RMLA  [6001041C]:      05C0(X)       ;SECTOR=23(D)
     RMSN  [60010420]:      2286(X)       ;
     RMOF  [60010424]:      1000(X)       ;FMT16
     RMDC  [60010428]:      0230(X)       ;DESIRED CYLINDER = 0560(D)
     RMMR2 [60010430]:      13FF(X)       ;TEST,BUS IN =3FF(X)
     RMER2 [60010434]:      0000(X)       ;
     RMEC1 [60010438]:      0836(X)       ;
     RMEC2 [6001043C]:      0000(X)       ;

(5)  MBA CHANNEL STATUS DUMP

     MBA_CSR:[20010000]      00000020(X);
     MBA_CR:[20010004]       00000000(X);
     MBA_SR:[20010008]       00002000(X);   DTCOMP
     MBA_VAR:[2001000C]      00000200(X);
     MBA_MAP(01):    00000000(X);
     MBA_BCNT:[20010010]     00000000(X);

(6)  ******** End of Hard error number 6 ********
```

Erläuterung:
(1) Überschrift mit Testname und Versionsnummer
(2) Durchlaufzahl, Test-, Subtest- u. Errornummer, Startzeit
(3) Fehlerart
(4) Name und Inhalt der Register
(5) MBA-Status
(6) Beendigungsmitteilung

6.8.5.8. Programmitteilungen

Bei Ausführung der Tests 10-13 (Sektion TIMING) werden die spezifizierten Zeiten und die aktuellen Zeiten ausgegeben.

6.8.5.9. Leistungsparameter des Testobjektes

entfällt

6.8.5.10. Aktionen bei unerwarteten Fehlern

entfällt

6.8.5.11. Testfolge bei mehreren Einheiten

entfällt

6.8.5.12. Einstellungen für Ablauf unter AFT

entfällt

6.8.5.13. Testsektionen

Das Programm enthält 8 Testsektionen, wobei jede mit dem Sektionsschalter in der Startkommandozeile ausgewählt und damit ausgeführt werden kann.

ACCESS

reserviert

ALL

Test 1-40

Es werden alle zur Verfügung stehenden Tests abgearbeitet.

DATA

Test 3-6,15,21-39

Die Datenübertragungs-Sektion führt alle möglichen Datenübertragungs-Kommandos aus. Damit wird die ordnungsgemäße Arbeit der HEADER-Logik (Kopfpositionierung) überprüft. Es wird die Adressierbarkeit aller Spuren und Sektoren nachgewiesen. Nachfolgend eine Liste der einzelnen Tests.

Gerät rücksetzen
illegale Funktion
Suchkommando
Suchkommando einschließlich Positionieren
unvollkommene Operation
Übertragung einschließlich Suchen
Übertragung einschließlich Positionieren
Interrupttest

Schreiben mit Prüfung Kopf und Daten
Schreiben mit Prüfung Daten
Übertragung mit letztem Sektor
Adreßüberlauf
falsche Adresse
falsches Format
Adreßvergleichsfehler
fehlerhafter Sektor
Spurauswahltest
Sektorauswahltest
Differenzen bei Zylinderpositionierung
Schreiben Kopf und Daten/Lesen Kopf und Daten
Schreiben/Lesen Daten Muster: ^X00
Schreiben/Lesen Daten Muster: 400 byte ^XFF, 112 byte ^XAA
Schreiben/Lesen Daten Muster: ^XAA
Schreiben/Lesen Daten Muster: ^X55

DEFAULT

Test 1-6,10-39

Diese Sektion arbeitet die wichtigsten Tests der einzelnen Sektionen hintereinander ab.

MANUAL

Test 07-09,40

Diese Sektion des Programms entspricht dem Dialogverkehr. Sie gestattet dem Operator die Eingabe variabler Funktionen. Nachfolgend eine Liste der einzelnen Tests.

PACK ACKNOWLEDGE (Gerätestatusabfrage)
READ IN PRESET (Lesevorbereitungstest)
Logischer Adreßstecker
Schreibschutzschaltertest

POSITION

Test 16-20

Diese Sektion enthält die Positioniertests. Sie dienen der vollständigen Prüfung des SERVO-Mechanismus.

SEEK TST

Test 16-20

Diese Sektion enthält die Positioniertests. Sie dienen der vollständigen Prüfung des SERVO-Mechanismus. Nachfolgend der Ablauf eines Positioniertests.

```
RECALIBRATE/SEEK TEST                 ;Neupositionierung über Zylinder 0 /Po-
                                      ;sitionieren
SEEK/SEEK (0-128)                     ;Positionieren zwischen Zylinder 0 und
                                      ;128
STEPPING SEEK TEST ( 0,1,2,4,8,16...) ;schrittweises Positionieren in Zweier-
                                      ;potenzen
CONVERGING/DIVERGING SEEK TEST        ;aufeinander zu / voneinander weg Posi-
```

```
                                       ;tionieren
OFFSET SEEK/RETURN TO CENTERLINE TEST  ;Versatzpositionierung / Nullpositionie-
                                       ;rung
```

TIMING

Test 10-14

Diese Sektion ermittelt die Zeitabläufe verschiedener Funktionen und vergleicht dann diese mit den spezifizierten Werten. Nachfolgend eine Liste mit den Funktionen und ihrer zeitlichen Zuordnung.

```
PACK SPEED  25,00ms +/- 7% 50Hz   ;Zeit für eine Plattenumdrehung
ONE CYL. SEEK  < 10ms             ;Zeit für nächsten Zylinder positionieren
ACCESS TIME    < 45ms             ;mittlere Zugriffszeit
MAX.SEEK       < 70ms             ;maximale Positionierzeit
```

6.8.6. Testbeschreibungen

6.8.6.1. Test 1: PROBE DEVICE REGISTERS Geräteregister ansprechen

Dieser Test prüft, ob alle Geräteregister lesbar sind.

Fehler:

Fehler 1: wird gemeldet, wenn INIT MBA nicht erfolgreich ablief.
Fehler 2: wird gemeldet, wenn das Löschen des MBA-Status nicht fehlerfrei war.
Fehler 3: wird gemeldet, wenn ein Geräteregister nicht lesbar ist.

6.8.6.2. Test 2: NOOP COMMAND Keine Operation

Dieser Test prüft, ob das Kommando NOOP richtig arbeitet. Er prüft dabei, ob lediglich das GO-Bit rückgesetzt wurde. Dazu werden folgende Testschritte ausgeführt:

1. aussenden MSBUS INIT
2. anweisen NOOP Kommando (und GO)
3. Auswertung von Registerstellen im Register RMCS1 und des Gerätestatus im Register RMDS.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID (Datenträger gültig) Fehler-Flags gesetzt sind. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn das GO-Bit auf Eins steht nach Ausführung von NOOP.
Fehler 3: wird gemeldet, wenn das Bit DVA (Gerät verfügbar) nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn außer DVA und GO noch andere Bits im Register RMCS1 gesetzt sind.
Fehler 5: wird gemeldet, wenn der Status DRY (Gerät bereit) im Register RMDS nicht gesetzt ist.
Fehler 6: wird gemeldet, wenn der Status DPR (Gerät vorhanden) im Register RMDS nicht gesetzt ist.

Fehler 7: wird gemeldet, wenn das Bit ERR im Register RMDS gesetzt ist.

6.8.6.3. Test 3: DRIVE CLEAR Gerät rücksetzen

Dieser Test schreibt alles Einsen in die Register RMER1, RMER2 und RMOF. Anschließend wird das Kommando DRIVE CLEAR ausgesendet und kontrolliert, ob dadurch sämtliche Bitstellen in den oben genannten Registern gelöscht wurden. Bevor jedoch die Register geladen werden, prüft dieser Test zuerst ob der Status MEDIUM ONLINE (Gerät online) gesetzt ist.

Fehler:

Fehler 1: wird gemeldet, wenn der Status MOL nicht gesetzt ist.
Fehler 2: wird gemeldet, wenn außer DVA und den Funktionsbits für DRIVE CLEAR noch andere Bits im Register RMCS1 gesetzt sind.
Fehler 3: wird gemeldet, wenn das Register RMER1 nach Ausführung von DRIVE CLEAR nicht vollständig gelöscht ist.
Fehler 4: wird gemeldet, wenn das Register RMOF durch DRIVE CLEAR nicht rückgesetzt werden konnte.
Fehler 5: wird gemeldet, wenn das Register RMER2 nach Ausführung von DRIVE CLEAR nicht vollständig gelöscht ist.

6.8.6.4. Test 4: ILLEGAL FUNCTION TEST Test auf illegale Funktion

Dieser Test prüft, ob ein illegaler Funktionscode den Fehler ILF hervorruft. Dabei werden folgende illegale Funktionen getestet (in HEX):
0A,0B,0D,0E,0F,10,11,12,13,16,17,1A,1B,1E,1F.
Als erstes springt dazu der Test die VOLUME VALID Routine an.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt sind. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn das Bit ILF bei versuchter Ausführung einer illegalen Funktion nicht gesetzt ist.

6.8.6.5. Test 5: SEARCH COMMAND Suchkommandotest

Bei diesem Test wird das Kommando SEARCH auf Zylinder 0, Spur 0, Sektor 0 ausgeführt.
Als erstes wird dazu das Kommando RECALIBRATE (Nullstellen) abgearbeitet, welches die Köpfe auf Zylinder 0 positioniert. War das erfolgreich, wird das Kommando SEARCH gestartet.
Anschließend wird das Register RMLA auf Sektor 0 kontrolliert.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt sind. Um welche Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.

Fehler 2: wird gemeldet, wenn das Kommando RECALIBRATE nicht fehlerfrei abgelaufen ist.
Fehler 3: wird gemeldet, wenn in einer bestimmten Zeit nach Start des Kommandos SEARCH das Bit ATA (Achtung) im Register RMDS nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn das Register RMLA nach Ausführung von SEARCH nicht die Sektornummer Null enthält.

6.8.6.6. Test 6: SEARCH COMMAND TEST WITH IMPLIED SEEK Suchen einschließlich Positionieren

Bei diesem Test wird das Kommando SEARCH auf Zylinder 10, Spur 0, Sektor 0 ausgeführt.
Als erstes wird dazu das Kommando RECALIBRATE (Nullstellen) abgearbeitet, welches die Köpfe auf Zylinder 0 positioniert. War das erfolgreich, wird das Kommando SEARCH mit <RMDC>=10 gestartet. Anschließend wird das Register RMLA auf Sektor 0 kontrolliert.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn das Kommando RECALIBRATE nicht fehlerfrei abgelaufen ist.
Fehler 3: wird gemeldet, wenn in einer bestimmten Zeit nach Start des Kommandos SEARCH das Bit ATA (Achtung) im Register RMDS nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn das Register RMLA nach Ausführung von SEARCH nicht die Sektornummer 0 enthält.

6.8.6.7. Test 7: PACK ACKNOWLEDGE TEST Gerätestatusabfrage

Dieser Test prüft, ob der Status VV (Datenträger gültig) als Ergebnis des PACK ACKNOWLEDGE gesetzt wurde. Die einzige Möglichkeit den Status VV zu löschen, ist das Gerät auszuschalten.
Als erstes wird dazu der Status VV abgefragt. Ist er gesetzt, wird der Bediener aufgefordert, das Gerät aus-und einzuschalten.
Anschließend wird der Status MOL (Gerät online) getestet. Wurde er nicht gesetzt, kommt es zum Fehler. Danach wird das Kommando PACK ACKNOWLEDGE ausgeführt und die Bits ERR und VV kontrolliert.

Fehler:

Fehler 1: wird gemeldet, wenn der Status MOL nach Einschalten des Gerätes nicht gesetzt ist.
Fehler 2: wird gemeldet, wenn nach Ausführung des Kommandos PACK ACKNOWLEDGE das Bit ERR im Register RMDS gesetzt ist.
Fehler 3: wird gemeldet, wenn nach Ausführung von PACK ACKNOWLEDGE der Status VV nicht gesetzt wurde.

6.8.6.8. Test 8: READ IN PRESET TEST
Lesevorbereitungstest

Dieser Test prüft, ob das Bit VV (Datenträger gültig) als Ergebnis des Kommandos READ IN PRESET gesetzt wurde, und ob die Register RMDC und RMDA gelöscht sind. Ebenfalls müssen die Bits ECCI, HCI und FMT16 im Register RMOF gelöscht sein.
Als erstes wird der Status VV abgefragt. Ist er gesetzt, wird der Bediener aufgefordert, das Gerät von Stellung offline in online zu bringen.
Anschließend wird der Status MOL (Gerät online) getestet. Ist er nicht gesetzt, kommt es zum Fehler. Danach werden die Register RMDC,RMDA und RMOF geladen,bevor das Kommando READ IN PRESET gestartet wird. Zum Schluß erfolgt die Kontrolle des Status ERR und VV und die Kontrolle der Register RMDC, RMDA, RMOF.

Fehler:

Fehler 1: wird gemeldet, wenn der Status MOL nach Einschalten des Gerätes nicht gesetzt ist.
Fehler 2: wird gemeldet, wenn der Status ERR nach Ausführung von READ IN PRESET gesetzt ist.
Fehler 3: wird gemeldet, wenn nach Ausführung von READ IN PRESET der Status VV nicht gesetzt ist.

6.8.6.9. Test 9: LOGICAL ADDRESS PLUGS TEST
Logischer Adreßstecker

Dieser Test prüft die Funktionsfähigkeit des logischen Adreßstecker.
Als erstes wird dazu das Kommando READ DATA im Diagnosemodus gestartet. Anschließend wird der Bediener aufgefordert, den Adreßstecker zu ziehen. Danach wird versucht, auf das Register RMCS1 zuzugreifen und kontrolliert, ob im Statusregister (SR) des MBA das Bit NED (nichtex. Gerät) gesetzt ist.
Anschließend wird der Bediener aufgefordert den Adreßstecker wieder anzustecken. Danach erfolgt die Kontrolle, ob der Status OPE (Bedienfehler) und LSC (Verlust Systemtakt) im Fehlerregister2 gesetzt sind.

Fehler:

Fehler 1: wird gemeldet, wenn der Status MOL nicht gesetzt ist.
Fehler 2: wird gemeldet, wenn der Status VV nicht gesetzt ist.
Fehler 3: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 4: wird gemeldet, wenn der Status NED im Statusregister des MBA nicht gesetzt ist.
Fehler 5: wird gemeldet, wenn das Bit OPE im Register RMER2 nicht gesetzt ist.
Fehler 6: wird gemeldet, wenn das Bit LSC im Register RMER2 nicht gesetzt ist.
Fehler 7: wird gemeldet, wenn die Routine VOLUME VALID Fehler-Flags gesetzt hat.

6.8.6.10. Test 10: PACK ROTATION SPEED TEST
Plattendrehzahltest

Diese Test mißt die Zeit für eine Plattenumdrehung. Es wird dazu das Kommando SEARCH auf Zylinder 0, Spur 0, Sektor 0 ausgeführt. Sobald ein Interrupt auftritt, wird nochmals das Kommando SEARCH zum gleichen Sektor ausgeführt. Das zweite SEARCH wird gemessen. Die Prozedur wird 10 mal wiederholt. Dann wird der Mittelwert berechnet. Die erwartete und die aktuelle Zeit werden ausgegeben.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn eine Fehlerbedingung nach dem ersten SEARCH-Kommando vorliegt.
Fehler 3: wird gemeldet, wenn die Umdrehungszeit außerhalb der Toleranzen liegt.

6.8.6.11. Test 11: ONE CYLINDER SEEK TIMING TEST Positionieren benachbarter Zylinder

Dieser Test führt eine Positionierung auf den nächsten Zylinder in beiden Richtungen durch. Es wird dabei geprüft, ob die Positionierzeit von 10ms nicht überschritten wird.
Als erstes wird dazu eine Vorwärtspositionierung zum Zylinder 1 ausgeführt. Anschließend erfolgt die Rückwärtspositionierung zum Zylinder 0. Dieser Zyklus wird 100 mal wiederholt. Aus diesen Zeiten wird dann die mittlere Positionierzeit berechnet.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Positionieren auf Zylinder 1 zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es beim Positionieren auf Zylinder 1 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es beim Positionieren auf Zylinder 0 zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es beim Rückwärtspositionieren auf Zylinder 0 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 6: wird gemeldet, wenn die Positionierung um einen Zylinder die Zeit von 10 ms überschreitet.

6.8.6.12. Test 12: ACCESS TIME MEASUREMENT TEST Mittlere Zugriffszeitmessung

Dieser Test führt eine Vorwärts-und Rückwärtspositionierung zwischen Zylinder 0 und Zylinder 136 aus. Dabei prüft der Test, ob die Zeit für jede Positionierung von 45 ms nicht überschritten wird.
Als erstes wird dazu eine Vorwärtspositionierung vom Zylinder 0 auf den Zylinder 136 ausgeführt. Die Zeit, die hierfür benötigt wird, wird abgespeichert. Anschließend erfolgt die Rückwärtspositionierung vom Zylinder 136 auf Zylinder 0. Dieser Zyklus wird 100 mal wiederholt. Aus allen Zugriffszeiten wird dann die mittlere Zugriffszeit errechnet.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Positionieren von Zylinder 0 auf Zylinder 136 zum Time-Out kommt.

Fehler 3: wird gemeldet, wenn es beim Positionieren auf Zylinder 136 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es beim Positionieren von Zylinder 136 auf Zylinder 0 zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es beim Positionieren auf Zylinder 0 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 6: wird gemeldet, wenn die Zugriffszeit von 45 ms überschritten wird.

6.8.6.13. Test 13: MAXIMUM SEEK TIMING TEST Maximale Positionierzeitmessung

Dieser Test führt eine Vorwärts-und Rückwärtspositionierung zwischen Zylinder 0 und Zylinder 560 aus. Dabei prüft der Test, ob die maximale Positionierzeit von 70 ms nicht überschritten wird.
Als erstes wird dazu eine Vorwärtspositionierung von Zylinder 0 auf Zylinder 560 ausgeführt. Die Zeit die hierfür benötigt wird, wird abgespeichert.
Anschließend erfolgt die Rückwärtspositionierung von Zylinder 560 auf Zylinder 0. Diese Zeit wird ebenfalls gerettet. Dieser Zyklus wird 100 mal wiederholt. Aus den einzelnen Positionierzeiten wird dann die aktuelle maximale Positionierzeit berechnet.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Positionieren vom Zylinder 0 auf Zylinder 560 zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es beim Positionieren auf Zylinder 560 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es beim Positionieren von Zylinder 560 auf Zylinder 0 zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es beim Positionieren auf Zylinder 0 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 6: wird gemeldet, wenn die maximale Positionierzeit von 70 ms überschritten wird.

6.8.6.14. Test 14: SKIP SECTOR ERROR (SSE) SKIP-Sektorfehler

Dieser Test prüft die SKIP-Sektorlogik. Der Test wird in drei Subtests ausgeführt.

Subtest 1

Im ersten Subtest werden mit dem Kommando WRITE HEADER UND DATA der Sektor 0 und 1 auf Zylinder 560,Spur 0 beschrieben. Der Sektor 0 enthält alles Nullen und der Sektor 1 alles Einsen.
Folgende Adreßworte werden geschrieben:

Adreßwort für Sektor 0: F230(X)
Adreßwort für Sektor 1: 1D230(X)

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler festgestellt hat.
Fehler 3: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn die Kontrolle des Register RMER1 negativ ausgeht.

Subtest 2

Dieser Subtest liest den Sektor 0 von Zylinder 560, Spur 0. Anschließend wird geprüft, ob der Status SSE gesetzt ist. Danach wird der Status SSE gelöscht, und das Bit SSEI (SKIP-SEKTOR-INHIBIT) und GO gesetzt. Damit erfolgt das Lesen des Sektors 1.

Fehler:

Fehler 1: wird gemeldet, wenn durch die Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen nicht übereinstimmt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es bei Ausführung von READ DATA zu Statusfehlern kommt.
Fehler 6: wird gemeldet, wenn der Sektor nicht richtig erhöht wurde bei SSE.

Subtest 3

Im dritten Subtest werden analog dem Subtest 1, mit dem Kommando WRITE HEADER AND DATA die Sektoren 0 und 1 auf Zylinder 560, Spur 0 beschrieben. Dabei wird aber für Sektor 0 folgendes Adreßwort geschrieben: D230(X).

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler feststellt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Register RMER1 negativ ausgeht.

6.8.6.15. Test 15: OPERATION INCOMPLETE TEST (OPI) Unvollkommene Operation

Dieser Test prüft, ob der Status OPI nicht gesetzt ist, wenn 3 Indeximpulse zusammentreffen während eines Lesekommandos.
Das Lesekommando wird gestartet ab Zylinder 0, Spur 0, Sektor 30 für 128 fortlaufende Sektoren. Es wird dann geprüft, ob OPI nicht gesetzt ist, oder irgendein anderer Fehler.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Löschen der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 4: wird gemeldet, wenn es bei Ausführung des Lesekommandos zu Statusfehlern kommt.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.

6.8.6.16. Test 16: RECALIBRATE/SEEK TEST Positionieren Null/ zur Position

Bei diesem Test wird das Kommando SEEK mit Zylinderadresse 560 ausgeführt. Anschließend wird das Kommando RECALIBRATE ausgeführt und geprüft, ob danach das Zylinderadreßregister die 0 enthält. Dieser Test wird 10 mal wiederholt.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Positionieren auf Zylinder 560 zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es bei der Ausführung des Kommandos SEEK zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es bei RECALIBRATE (Nullstellen) zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es bei Ausführung von RECALIBRATE zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

6.8.6.17. Test 17: SEEK/SEEK-TEST Positionieren vorwärts/rückwärts

Dieser Test führt ein Vorwärtspositionieren zum Zylinder 128 aus. Danach wird das Zylinderadreßregister auf seine Richtigkeit kontrolliert. Anschließend erfolgt ein Rückwärtspositionieren zum Zylinder 0 und danach die Kontrolle auf Fehlerbedingungen. Dieser Zyklus wird 10 mal wiederholt.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Positionieren auf Zylinder 0 zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es beim Positionieren auf Zylinder 0 zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es beim Positionieren auf Zylinder 128 zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es beim Positionieren auf Zylinder 128 zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

6.8.6.18. Test 18: STEPPING SEEK TEST Schrittweise Positionieren

Dieser Test führt die Positionierung zu den Zylindern 0,1,2,4,8,16,32,64,128, 256,512 durch.
Anschließend wird positioniert zu 256,128,64,32 u.s.w.
Nach jedem Positionierkommando erfolgt eine Testung auf Statusfehler, den korrekten Inhalt des Zylinderadreßregisters und Geräte Time-Outs.

Fehler:

Fehler 1: wird mitgeteilt, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird mitgeteilt, wenn es bei der Ausführung von SEEK zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es bei der Ausführung von SEEK zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

6.8.6.19. Test 19: CONVERGING/DIVERGING SEEK TEST Aufeinander zu/voneinander weg Positionieren

Dieser Test führt aufeinander zu/voneinander weg Positionierungen aus.
Gestartet wird mit einer Positionierung von Zylinder 0 bis zum letzten Zylinder
Dann erfolgt eine Positionierung vom letzten Zylinder bis zum ersten.
Anschließend vom ersten Zylinder zum letzten Zylinder -1 u.s.w.,bis die Mitte der Platte erreicht ist. Danach wird das voneinander weg Positionieren durchgeführt. Nach jedem SEEK wird auf Statusfehler und Richtigkeit des Zylinderadreßregisters geprüft.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es bei einer Positionierung beginnend von Zylinder 0 bis zum letzten Zylinder zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn es bei einer Positionierung beginnend von Zylinder 0 bis zum letzten Zylinder zum Time-Out kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 4: wird gemeldet, wenn es bei einer Positionierung beginnend von Zylinder 560 bis Zylinder 0 zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn es bei einer Positionierung beginnend von Zylinder 560 bis Zylinder 0 zum Statusfehler kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 6: wird gemeldet,bei Time-Out.
Fehler 7: wird gemeldet,bei Statusfehlern.
Fehler 8: wird gemeldet,bei Time-Out.
Fehler 9: wird gemeldet,bei Statusfehlern.

6.8.6.20. Test 20: OFFSET/RETURN TO CENTERLINE Offset- und Normalpositioniertest

Dieser Test prüft, ob die Kommandos OFFSET und RETURN TO CENTERLINE richtig ablaufen. Zuerst wird dazu das Kommando RECALIBRATE abgearbeitet. Damit werden die Köpfe auf Zylinder 0 positioniert. Danch läuft der Test in 2 Subtests ab.

Subtest 1

Die Offsetadresse wird in das Register RMOF geladen und das Kommando OFFSET gestartet. Es erfolgt danach ein Test auf mögliche Fehler. Dieser Subtest wird für alle möglichen Offsetadressen wiederholt.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Kommando OFFSET zum Time-Out kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung des Kommandos OFFSET zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 3: wird gemeldet, wenn nach Ausführung von OFFSET der Offset-Modus nicht eingeschaltet ist.

Subtest 2

Dieser Subtest prüft, ob das Kommando RETURN TO CENTERLINE von jeder möglichen Offsetposition ausgeführt werden kann.
Als erstes wird dazu die Funktion OFFSET mit geladener Offsetadresse im Register RMOF gestartet. Danach wird auf Fehler geprüft.
Anschließend wird das Kommando RETURN TO CENTERLINE ausgeführt. Danach wird ebenfalls auf Fehlerbedingungen geprüft.
Außerdem wird geprüft, ob das Register RMOF danach gelöscht ist. Dieser Subtest wird für jede Offsetadresse wiederholt.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Kommando OFFSET zum Time-Out kommt.
Fehler 2: wird gemeldet, wenn es nach Ausführung von OFFSET zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 3: wird gemeldet, wenn es beim Kommando RETURN TO CENTERLINE zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn es nach Ausführung von RETURN TO CENTERLINE zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.
Fehler 5: wird gemeldet, wenn der Offsetmodus nach RETURN TO CENTERLINE nicht gelöscht ist.

6.8.6.21. Test 21: IMPLIED SEARCH TEST Übertragung einschließlich Suchen

Dieser Test ist in zwei Subtests unterteilt.

Testparameter:

```
Zylinder  = 560
Spur      = 0
Sektor    = 0
Bytezahl  = 524
Daten     =   4 byte HEADER
            512 byte alles Nullen
              4 byte HEADER
              4 byte Einsen
```

Subtest 1

Dieser Subtest führt das Kommando WRITE HEADER AND DATA ab Zylinder 560, Spur 0, Sektor 0 aus.

Fehler:

Fehler 1: wird gemeldet, wenn durch die Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler feststellt.
Fehler 4: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Fehlerregisters 1 negativ ausgeht.

Subtest 2

Dieser Subtest liest 532 byte von Zylinder 560, Spur 0, Sektor 0. Es ist das erste Lesekommando, welches mehr als einen Sektor liest.

Fehler:

Fehler 1: wird gemeldet, wenn durch die Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn die Routine STATUS CHECK Fehler feststellt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Fehlerregisters 1 negativ ausgeht.

6.8.6.22. Test 22: IMPLIED SEEK TEST Übertragung einschließlich Positionieren

Der Test führt eine Übertragung einschließlich Positionieren durch. Er ist in zwei Subtests unterteilt.

Testparameter:

```
Zylinder  = 559
Spur      = 13
Sektor    = 30
Bytezahl  = 524
Daten     =   4 byte HEADER
            512 byte alles Nullen
              4 byte HEADER
              4 byte Einsen
```

Subtest 1

Dieser Subtest führt das Kommando WRITE HEADER AND DATA beginnend ab Zylinder 559, Spur 13, Sektor 30 aus. Es werden dabei 524 byte geschrieben.

Fehler:

Fehler 1: wird gemeldet, wenn durch die Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler festgestellt hat.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Fehlerregisters 1 negativ ausgeht.

Subtest 2

Dieser Test führt das Kommando READ HEADER AND DATA beginnend ab Zylinder 559, Spur 13, Sektor 30. Es werden dabei 524 byte gelesen.

Fehler:

Fehler 1: wird gemeldet, wenn durch die Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler feststellt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Fehlerregister 1 negativ ausgeht.
Fehler 6: wird gemeldet, wenn der Vergleich der geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

6.8.6.23. Test 23: INTERRUPT TEST Interrupttest

Dieser Test prüft, ob ein Interrupt auftritt,nach einem kompletten Datentransport-oder Positionierkommando. Der Test benutzt dazu die Kommandos WRITE DATA, READ DATA und SEEK mit gesetzter Unterbrecherlaubnis.
Außerdem erfolgt eine Fehlerkontrolle. Der Test ist in drei Subtests unterteilt.

Subtest 1

Dieser Subtest prüft, ob ein Interrupt auftritt,nach einem kompletten WRITE DATA-Kommando mit gesetzter Unterbrecherlaubnis.

Fehler:

Fehler 1: wird gemeldet,bei Supervisor-Fehler
Fehler 2: wird gemeldet, wenn kein Interrupt auftritt, nach Ausführung von WRITE DATA.
Fehler 3: wird gemeldet, wenn es bei WRITE DATA zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

Subtest 2

Dieser Subtest prüft, ob ein Interrupt auftritt,nach einem kompletten READ DATA-Kommando mit gesetzter Unterbrecherlaubnis.

Fehler:

Fehler 1: wird gemeldet,bei Supervisor-Fehler
Fehler 2: wird gemeldet, wenn kein Interrupt auftritt, nach Ausführung von READ DATA.
Fehler 3: wird gemeldet, wenn es bei READ DATA zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

Subtest 3

Dieser Subtest prüft, ob ein Interrupt auftritt, wenn die Achtungleitung aktiv ist. Dazu führt der Test das Kommando SEEK aus, welches ein Achtung hervorruft.

Fehler:

Fehler 1: wird gemeldet,bei Supervisor-Fehler.
Fehler 2: wird gemeldet, wenn kein Interrupt auftritt, nach Ausführung von SEEK.
Fehler 3: wird gemeldet, wenn es bei SEEK zu Statusfehlern kommt. Das Bit ERR im Register RMDS ist gesetzt.

6.8.6.24. Test 24: WRITE CHECK HEADER + DATA-TEST Schreiben mit Prüfung Kopf und Daten

Testparameter:

```
Zylinder = 560
Spur     = 4
Sektor   = 7
Bytezahl = 524
Daten    =   4 byte Header
            10 Worte 14E4(16)
            20 byte alles Einsen
            20 byte alles Nullen
            20 byte 55(16)
            20 byte AA(16)
            32 byte float. Null
            32 byte float. Eins
           174 Worte FF00(X)
             4 byte Header
             4 byte 14E5(X)
```

Subtest 1

Dieser Subtest führt das Kommando WRITE HEADER AND DATA aus.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es nach Ausführung von WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei WRITE HEADER AND DATA zum Time-Out kommt.

Subtest 2

Dieser Subtest führt das Kommando READ HEADER AND DATA aus.
Anschließend werden die gelesenen Daten mit den im vorhergehenden Subtest geschriebenen Daten verglichen.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von READ HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei READ HEADER AND DATA zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

Subtest 3

Dieser Subtest führt das Kommando WRITE CHECK HEADER AND DATA aus, unter Benutzung der Daten, welche in Subtest 1 geschrieben wurden.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von WRITE CHECK HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei Ausführung von WRITE CHECK HEADER AND DATA zum Time-Out kommt.

6.8.6.25. Test 25: WRITE CHECK DATA TEST
Schreiben mit Prüfung Daten

Testparameter:

```
Zylinder  =560
Spur      =4
Sektor    =7
Bytezahl  = 516
Daten     =  10 Worte 14E4(16)
             20 byte alles Einsen
             20 byte alles Nullen
             20 byte 55(16)
             20 byte AA(16)
             32 byte float. Null
             32 byte float. Eins
            174 Worte FF00(X)
              4 byte 14E5(X)
```

Subtest 1

Dieser Subtest führt das Kommando WRITE DATA aus.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von WRITE DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei WRITE DATA zum Time-Out kommt.

Subtest 2

Dieser Subtest führt das Kommando READ DATA aus. Anschließend werden die gelesenen Daten mit den im vorhergehenden Subtest geschriebenen Daten verglichen.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von READ DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei READ DATA zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn der Vergleich von geschriebenen mit den gelesenen Daten nicht übereinstimmt.

Subtest 3

Dieser Subtest führt das Kommando WRITE CHECK DATA unter Benutzung der Lese-und Schreibdaten der vorhergehenden zwei Subtests durch.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von WRITE CHECK DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei WRITE CHECK DATA zum Time-Out kommt.

6.8.6.26. Test 26: LAST SECTOR TRANSFERRED (LST) Letzter Sektor übertragen

Dieser Test schreibt 512 byte ab Zylinder 560, Spur 13, Sektor 30. Danach wird geprüft, ob der Status LST gesetzt ist.
Anschließend erfolgt das Lesen von dieser Plattenadresse und es wird ebenfalls geprüft, ob danach der Status LST gesetzt wurde. Dieser Test ist in zwei Subtest untergliedert.

Subtest 1

Im ersten Subtest erfolgt das Schreiben der 512 byte und die Kontrolle auf den Status LST.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung des Kommandos WRITE DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei WRITE DATA zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn nach Kommandoausführung der Status LST nicht gesetzt ist.

Subtest 2

Bei diesem Subtest erfolgt das Lesen der 512 byte und ebenfalls die Kontrolle auf den Status LST.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es bei Ausführung von READ DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es bei READ DATA zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn nach Kommandoausführung der Status LST nicht gesetzt ist.
Fehler 5: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen nicht übereinstimmt.

6.8.6.27. Test 27: ADDRESS OVERFLOW ERROR (AOE) Adreßüberlauftest

Dieser Test schreibt 513 byte ab Zylinder 560, Spur 13, Sektor 30. Danach wird geprüft, ob die beiden Bits AOE und LST gesetzt sind.
Anschließend wird vom Zylinder 560, Spur 13, Sektor 30 gelesen und ebenfalls kontrolliert, ob danach AOE und LST gesetzt sind. Dieser Test ist in zwei Subtests unterteilt.

Subtest 1

In diesem Subtest erfolgt das Schreiben der 513 byte und die Kontrolle auf AOE und LST.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn nach Ausführung von WRITE DATA der Status AOE nicht gesetzt ist.
Fehler 3: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn es bei WRITE DATA zu Statusfehlern kommt.
Fehler 5: wird gemeldet, wenn es bei WRITE DATA zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn der Status LST nicht gesetzt ist.

Subtest 2

Bei diesem Subtest erfolgt das Lesen der 513 byte und ebenfalls die Kontrolle auf Status AOE und LST.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn nach Ausführung von READ DATA der Status AOE nicht gesetzt ist.
Fehler 3: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn es bei READ DATA zu Statusfehlern kommt.
Fehler 5: wird gemeldet, wenn es bei READ DATA zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

6.8.6.28. Test 28: INVALID ADDRESS ERROR (IAE) Falsche Adresse

Dieser Test ist in drei Subtests unterteilt.

Subtest 1

Im ersten Subtest wird versucht ab Zylinder 561, Spur 0, Sektor 0 das Kommando WRITE HEADER AND DATA auszuführen. Der Bytezähler beträgt dabei 512. Es wird kontrolliert, ob hierbei der Status IAE (falsche Adresse) gesetzt wurde.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn der Status IAE nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn das Bit MBEXC (EXCEPTION) im Statusregister des MBA nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn außer dem Statusfehler IAE noch andere Fehlerbits gesetzt sind.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.

Subtest 2

Im zweiten Subtest wird versucht ab Zylinder 560, Spur 0, Sektor 0, das Kommando WRITE DATA auszuführen. Der Bytezähler beträgt 512. Es wird dann kontrolliert, ob ebenfalls der Status IAE gesetzt wurde.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn der Status IAE nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn das Bit MBEXC (EXCEPTION) im Statusregister des MBA nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn außer dem Statusfehler IAE noch andere Fehlerbits gesetzt sind.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.

Subtest 3

Im dritten Subtest wird versucht ab Zylinder 560, Spur 0, Sektor 32 das Kommando READ HEADER AND DATA auszuführen. Der Bytezähler beträgt 512. Es wird dann kontrolliert, ob der Status IAE gesetzt wird.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn der Status IAE nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn das Bit MBEXC (EXCEPTION) im Statusregister des MBA nicht gesetzt ist.
Fehler 4: wird gemeldet, wenn außer dem Statusfehler IAE auch noch andere Fehlerbits gesetzt sind.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.

6.8.6.29. Test 29: FORMAT ERROR (FER)
Formatfehler

Dieser Test ist in zwei Subtests unterteilt.

Subtest 1

Im ersten Subtest wird versucht 512 byte ab Zylinder 560, Spur 0, Sektor 0 im 18-bit-Modus aufzuzeichnen. Der Header ist dabei im 16-bit-Modus formatiert. Anschließend wird kontrolliert, ob der Status FER (Formatfehler) im Register RMER1 gesetzt ist und ob im Statusregister des MBA das Bit MBEXC (EXCEPTION) gesetzt ist.

Fehler:

Fehler 1: wird gemeldet, wenn das Formatfehlerbit FER nicht gesetzt wurde.
Fehler 2: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn im Fehlerregister 1 außer dem Fehlerbit FER noch andere Fehler gesetzt sind.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.

Subtest 2

Dieser Subtest verläuft analog dem ersten. Nur für das Kommando WRITE DATA wird das Kommando READ DATA benutzt.

Fehler:

Fehler 1: wird gemeldet, wenn das Formatfehlerbit FER nicht gesetzt wurde.
Fehler 2: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn im Fehlerregister 1 außer dem Fehlerbit FER noch andere Fehler gesetzt sind.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.

6.8.6.30. Test 30: HEADER COMPARE ERROR (HCE)
Adreßvergleichsfehler

Der Test 30 ist in 4 Testabschnitte unterteilt.

Umformatierung Sektor 1

In diesem Testabschnitt wird der Zylinder 560, Spur 0, Sektor 1 als Sektor 0 formatiert.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.

Subtest 1

In diesem Subtest wird versucht vom Zylinder 560, Spur 0, Sektor 1 zu lesen. Dabei muß es zum Statusfehler HCE kommen.

Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.

Subtest 1

In diesem Subtest wird versucht vom Zylinder 560, Spur 0, Sektor 1 zu lesen. Dabei muß es zum Statusfehler HCE kommen.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn der Status HCE nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC (EXCEPTION) nicht gesetzt wurde.
Fehler 4: wird gemeldet, wenn im Register RMER1 außer dem Fehlerbit HCE noch andere Fehler gesetzt sind.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.

Subtest 2

Bei diesem Subtest wird versucht auf Zylinder 560, Spur 0, Sektor 1 zu schreiben. Dabei muß ebenfalls der Status HCE gesetzt sein.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn der Status HCE nicht gesetzt wurde.
Fehler 3: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC (EXCEPTION) nicht gesetzt wurde.
Fehler 4: wird gemeldet, wenn im Register RMER1 außer dem Fehlerbit HCE noch andere Fehler gesetzt sind.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.

Rückformatierung Sektor 1

In diesem Test wird der Sektor 1 wieder richtig formatiert.

Fehler:

Fehler 1: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 2: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es zum Time-Out kommt.

6.8.6.31. Test 31: BAD SECTOR ERROR TEST (BSE) Fehlerhafter Sektor

Der Test 31 ist in 4 Testabschnitte unterteilt.

Umformatierung Sektor 1

In diesem Testabschnitt wird der Zylinder 560, Spur 0, Sektor 1 formatiert. Die BAD-Sektorflags werden gelöscht.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.

Subtest 1

In diesem Subtest wird versucht vom Zylinder 560, Spur 0, Sektor 1 zu lesen. Dabei muß es zum Statusfehler BSE kommen.

Fehler:

Fehler 5: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 6: wird gemeldet, wenn der Status BSE nicht gesetzt wurde.
Fehler 7: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC (EXCEPTION) nicht gesetzt wurde.
Fehler 8: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 9: wird gemeldet, wenn im Register RMER1 Fehlerbits gesetzt wurden.

Subtest 2

Bei diesem Subtest wird versucht auf Zylinder 560, Spur 0, Sektor 1 zu schreiben. Dabei muß ebenfalls der Status BSE gesetzt sein.

Fehler:

Fehler 10: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 11: wird gemeldet, wenn der Status BSE nicht gesetzt ist.
Fehler 12: wird gemeldet, wenn im Statusregister des MBA das Bit MBEXC (EXCEPTION) nicht gesetzt wurde.
Fehler 13: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 14: wird gemeldet, wenn im Register RMER1 Fehlerbits gesetzt sind.

Rückformatierung Sektor 1

In diesem Test wird der Sektor 1 wieder richtig formatiert.

Fehler:

Fehler 15: wird gemeldet, wenn es beim Aufbau der MAP-Register zu Fehlern kommt.
Fehler 16: wird gemeldet, wenn die Routine STATUS CHECK Statusfehler feststellt.

Fehler 17: wird gemeldet, wenn es zum Time-Out kommt.

6.8.6.32. Test 32: HEAD SELECTION TEST Spurauswahltest

Dieser Test prüft die Kopfauswahllogik.
Als erstes wird dazu das Kommando WRITE HEADER AND DATA ab Zylinder 560, Sektor 0, Spur 0 ausgeführt.
Anschließend folgt das Kommando READ HEADER AND DATA. Die Daten werden dann miteinander verglichen. Dieser Test wird für die Spuren 1 bis 13 wiederholt.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden. Um welchen Fehler es sich handelt, ist aus der Fehlerausschrift zu ersehen.
Fehler 2: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn es beim Kommando READ HEADER AND DATA zu Statusfehlern kommt.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

6.8.6.33. Test 33: SECTOR SELECTION TEST Sektorauswahltest

Dieser Test prüft die Sektorauswahllogik,
Als erstes wird dazu das Kommando WRITE HEADER AND DATA ab Zylinder 560, Spur 0, Sektor 0 ausgeführt.
Anschließend erfolgt das Kommando READ HEADER AND DATA. Die Daten werden dann miteinander verglichen. Dieser Test wird wiederholt für die Sektoren 1 bis 30.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Kommando WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 3: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 4: wird gemeldet, wenn es beim Kommando READ HEADER AND DATA zu Statusfehlern kommt.
Fehler 5: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 6: wird gemeldet, wenn der Vergleich der geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

6.8.6.34. Test 34: DIFFERENT LINE TEST Differenzen bei Zylinderpositionierung

Dieser Test führt das Kommando READ HEADER AND DATA auf folgende Zylinder 0,1,2,4,8,16,32,64,128,256,512, Sektor 0, Spur 0 aus. Die Daten sind dabei die

Zylinderadressen.
Als erstes führt dazu der Test jeweils das Kommando RECALIBRATE (Nullstellen) durch.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn die Funktion RECALIBRATE fehlerhaft arbeitet.
Fehler 3: wird gemeldet, wenn durch die Routine STATUS CHECK nach Start von READ HEADER AND DATA Statusfehler gesetzt wurden.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn eine falsche Adresse gelesen wurde.

6.8.6.35. Test 35: WRITE $ READ HEADER $ DATA-TEST
Schreiben $ Lesen Kopf $ Daten

Dieser Test prüft die Kommandos WRITE HEADER AND DATA und READ HEADER AND DATA.
Dieser Test ist in zwei Subtests unterteilt.
Die Testparameter sind für beide Subtests gleich.

Zylinder = 560
Spur = 0
Sektor = 0
Bytezahl = 516
Daten = alles Nullen

Subtest 1

Dieser Subtest führt das Kommando WRITE HEADER AND DATA ab Zylinder 560, Sektor 0, Spur 0 aus.

Fehler:

Fehler 1: wird gemeldet, wenn die Routine VOLUME VALID Fehler-Flags gesetzt hat.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es bei Ausführung des Kommandos WRITE HEADER AND DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Registers RMER1 negativ ausgeht.

Subtest 2

Dieser Subtest führt das Kommando READ HEADER AND DATA aus.
Anschließend werden die gelesenen Daten mit den geschriebenen Daten verglichen.
Dieser Subtest wird nicht ausgeführt, wenn im Subtest 1 ein Fehler festgestellt wurde.

Fehler:

Fehler 1: wird gemeldet, wenn in der Routine VOLUME VALID Fehler-Flags gesetzt wurden.

Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es bei Ausführung von READ HEADER AND DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Registers RMER1 negativ ausgeht.
Fehler 6: wird gemeldet, wenn der Vergleich von geschriebenen Daten mit den gelesenen Daten nicht übereinstimmt.

6.8.6.36. Test 36: WRITE/READ DATA TESTS (DATA = ^X00) Schreiben/Lesen Daten (Muster = ^X00)

Dieser Test prüft die Kommandos WRITE DATA und READ DATA.
Der Test ist in zwei Subtests unterteilt.
Die folgenden Testparameter gelten für beide Subtests.

Zylinder	=	560
Spur	=	0
Sektor	=	0
Bytezahl	=	512
Daten	=	alles Nullen

Dieser Test ist ähnlich dem Test 35, außer daß die Kopfvergleichslogik benutzt wird. Der benutzte Sektor wurde im vorhergehenden Test formatiert.

Subtest 1

Dieser Subtest schreibt 512 byte Nullen ab Zylinder 560, Spur 0, Sektor 0.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zu Fehlern kommt.
Fehler 3: wird gemeldet, wenn es bei Ausführung von WRITE DATA zu Statusfehlern kommt.
Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Registers RMER1 negativ ausfällt.

Subtest 2

Dieser Subtest liest 512 byte von Zylinder 560, Spur 0, Sektor 0.
Anschließend werden die Daten verglichen mit den vorher geschriebenen.

Fehler:

Fehler 1: wird gemeldet, wenn bei der Routine VOLUME VALID Fehler-Flags gesetzt wurden.
Fehler 2: wird gemeldet, wenn es beim Aufbau der MAP-Register zum Fehler kommt.
Fehler 3: wird gemeldet, wenn es bei Ausführung von READ DATA zu Statusfehlern kommt.

Fehler 4: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 5: wird gemeldet, wenn die Kontrolle des Registers RMER1 negativ ausgeht.
Fehler 6: wird gemeldet, wenn der Vergleich der geschriebenen Daten mit den gelesenen nicht übereinstimmt.

6.8.6.37. Test 37: WRITE/READ DATA-TESTS (DATA = ^XFF, ^XAA) Schreiben/Lesen Daten (Muster = ^XFF, ^XAA)

Der Test 37 verläuft analog dem Test 36. Es hat sich hierbei nur das Datenmuster geändert. Bei diesem Test werden 400 byte Einsen und 112 byte AA(X) übertragen. Die Fehlererläuterung des Testes 36 trifft für diesen Test ebenfalls zu.

6.8.6.38. Test 38: WRITE/READ DATA TESTS (DATA = ^XAA) Schreiben/Lesen Daten (Muster = ^XAA)

Der Test 38 verläuft analog dem Test 36. Es hat sich hierbei nur das Datenmuster geändert. Bei diesem Test werden 512 byte AA(X) übertragen.
Die Fehlererläuterung des Testes 36 trifft für diesen Test ebenfalls zu.

6.8.6.39. Test 39: WRITE/READ DATA TESTS (DATA = ^X55) Schreiben/Lesen Daten (Muster = ^X55)

Der Test 39 verläuft analog dem Test 36. Es hat sich hierbei nur das Datenmuster geändert. Bei diesem Test werden 512 byte 55(X) übertragen.
Die Fehlererläuterung des Testes 36 trifft für diesen Test ebenfalls zu.

6.8.6 40. Test 40: WRITE PROTECT SWITCH TEST Schreibschutzschaltertest

Dieser Test prüft die Funktionsfähigkeit des Schreibschutzschalters.
Dazu schreibt der Test alles Einsen ab Zylinder 560, Spur 0, Sektor 0.
Anschließend wird der Bediener aufgefordert, den Schreibschutzschalter einzuschalten. Danach versucht der Test alles Nullen auf den gleichen Sektor zu schreiben. Dabei muß der Status WLE gesetzt sein.
Anschließend kontrolliert der Test, ob die Daten sich nicht verändert haben.
Zum Schluß wird der Bediener aufgefordert, den Schreibschutzschalter wieder auszuschalten.

Fehler:

Fehler 1: wird gemeldet, wenn in der Routine STATUS CHECK Statusfehler festgestellt wurden.
Fehler 2: wird gemeldet, wenn es zum Time-Out kommt.
Fehler 3: wird gemeldet, wenn bei gesetztem Schreibschutzschalter und Anweisung von WRITE DATA der Status WLE nicht gesetzt wurde.
Fehler 4: wird gemeldet, wenn das Bit MBEXC im Statusregister des MBA nicht gesetzt ist.

Fehler 5: wird gemeldet, wenn außer dem Fehlerbit WLE noch andere Fehle gesetzt sind.
Fehler 6: wird gemeldet, wenn es bei der zweiten Anweisung von WRITE DATA Time-Out kommt.
Fehler 7: wird gemeldet, wenn bei Ausführung von READ DATA zu Statusfe kommt.
Fehler 8: wird gemeldet, wenn es bei READ DATA zum Time-Out kommt.
Fehler 9: wird gemeldet, wenn der Vergleich der geschriebenen Daten mit gelesenen Daten nicht übereinstimmt.

6.8.7. Auflistung der Tests

Test 1: PROBE DEVICE REGISTERS
Geräteregister ansprechen
Test 2: NOOP COMMAND
Keine Operation
Test 3: DRIVE CLEAR
Gerät rücksetzen
Test 4: Test auf illegale Funktion
Illegale Funktion
Test 5: SEARCH COMMAND
Suchkommandotest
Test 6: SEARCH COMMAND TEST WITH IMPLIED SEEK
Suchen einschließlich Positionieren
Test 7: PACK ACKNOWLEDGE TEST
Gerätestatusabfrage
Test 8: READ IN PRESET TEST
Lesevorbereitungstest
Test 9: LOGICAL ADDRESS PLUGS TEST
Logischer Adreßstecker
Test 10: PACK ROTATION SPEED TEST
Plattendrehzahltest
Test 11: ONE CYLINDER SEEK TIMING TEST
Positionieren benachbarter Zylinder
Test 12: ACCESS TIME MEASUREMENT TEST
Mittlere Zugriffszeitmessung
Test 13: MAXIMUM SEEK TIMING TEST
Maximale Positionierzeitmessung
Test 14: SKIP SECTOR ERROR (SSE)
SKIP-Sektorfehler
Test 15: OPERATION INCOMPLETE TEST
Unvollkommene Operation
Test 16: RECALIBRATE/SEEK TEST
Positionieren Null/zur Position
Test 17: SEEK/SEEK TEST
Positionieren vorwärts/rückwärts
Test 18: STEPPING SEEK TEST
Schrittweise Positionieren
Test 19: CONVERGING/DIVERGING SEEK TEST
Aufeinander zu/voneinander weg Positionieren
Test 20: OFFSET/RETURN TO CENTERLINE
Offset-und Normlpositioniertest
Test 21: IMPLIED SEARCH TEST
Übertragung einschließlich Suchen
Test 22: IMPLIED SEEK TEST
Übertragung einschließlich Positionieren

Test 23:	INTERRUPT TEST Interrupttest
Test 24:	WRITE CHECK HEADER + DATA TEST Schreiben mit Prüfung Kopf und Daten
Test 25:	WRITE CHECK DATA TEST Schreiben mit Prüfung Daten
Test 26:	LAST SECTOR TRANSFERRED (LST) Letzter Sektor übertragen
Test 27:	ADDRESS OVERFLOW ERROR (AOE) Adreßüberlauftest
Test 28:	INVALID ADDRESS ERROR (IAE) Falsche Adresse
Test 29:	FORMAT ERROR (FER) Formatfehler
Test 30:	HEADER COMPARE ERROR (HCE) Adreßvergleichsfehler
Test 31:	BAD SECTOR ERROR (BSE) Fehlerhafter Sektor
Test 32:	HEAD SELECTION TEST Spurauswahltest
Test 33:	SECTOR SELECTION TEST Sektorauswahltest
Test 34:	DIFFERENT LINE TEST Differenzen bei Zylinderpositionierung
Test 35:	WRITE & READ HEADER & DATA TEST Schreiben & Lesen Kopf & Daten
Test 36:	WRITE/READ DATA TESTS (DATA = ^X00) Schreiben/Lesen Daten (Muster = ^X00)
Test 37:	WRITE/READ DATA TESTS (DATA = ^XFF, ^XAA) Schreiben/Lesen Daten (Muster = ^XFF, ^XAA)
Test 38:	WRITE/READ DATA TESTS (DATA = ^XAA) Schreiben/Lesen Daten (Muster = ^XAA)
Test 39:	WRITE/READ DATA TESTS (DATA = ^X55) Schreiben/Lesen Daten (Muster = ^X55)
Test 40:	WRITE PROTECT SWITCH TEST Schreibschutzschaltertest

Abkürzungsverzeichnis

BLP	bestückte Leiterplatte
BR	Busanforderungsstufe
CR	Carriage Return (Wagenrücklauf)
DP-BLP	Zweikanal-BLP
DS	Diagnosesupervisor
GKS	Graphisches Kernsystem
IC-BLP	Interfacesteuerung-BLP
MBA	MSBUS-Adapter
RG-BLP	Register-BLP
SS-BLP	Positionier-Such-BLP
TR	Übertragungsanforderungsstufe